

## По вопросам продаж и поддержки обращайтесь:

Архангельск +7 (8182) 45-71-35	Кемерово +7 (3842) 21-56-70	Новосибирск +7 (383) 235-95-48	Сочи +7 (862) 279-22-65
Астрахань +7 (8512) 99-46-80	Киров +7 (8332) 20-58-70	Омск +7 (381) 299-16-70	Ставрополь +7 (8652) 57-76-63
Барнаул +7 (3852) 37-96-76	Краснодар +7 (861) 238-86-59	Орел +7 (4862) 22-23-86	Сургут +7 (3462) 77-96-35
Белгород +7 (4722) 20-58-80	Красноярск +7 (391) 989-82-67	Оренбург +7 (3532) 48-64-35	Тверь +7 (4822) 39-50-56
Брянск +7 (4832) 32-17-25	Курск +7 (4712) 23-80-45	Пенза +7 (8412) 23-52-98	Томск +7 (3822) 48-95-05
Владивосток +7 (4232) 49-26-85	Липецк +7 (4742) 20-01-75	Пермь +7 (342) 233-81-65	Тула +7 (4872) 44-05-30
Волгоград +7 (8442) 45-94-42	Магнитогорск +7 (3519) 51-02-81	Ростов-на-Дону +7 (863) 309-14-65	Тюмень +7 (3452) 56-94-75
Екатеринбург +7 (343) 302-14-75	Москва +7 (499) 404-24-72	Рязань +7 (4912) 77-61-95	Ульяновск +7 (8422) 42-51-95
Ижевск +7 (3412) 20-90-75	Мурманск +7 (8152) 65-52-70	Самара +7 (846) 219-28-25	Уфа +7 (347) 258-82-65
Казань +7 (843) 207-19-05	Наб.Челны +7 (8552) 91-01-32	Санкт-Петербург +7 (812) 660-57-09	Хабаровск +7 (421) 292-95-69
Калуга +7 (4842) 33-35-03	Ниж.Новгород +7 (831) 200-34-65	Саратов +7 (845) 239-86-35	Челябинск +7 (351) 277-89-65
			Ярославль +7 (4852) 67-02-35

сайт: [moxa.pro-solution.ru](http://moxa.pro-solution.ru) | эл. почта: [mhk@pro-solution.ru](mailto:mhk@pro-solution.ru)

телефон: 8 800 511 88 70

# Руководство пользователя Smartio C168H/HS

Данный продукт поставляется по лицензионному соглашению и может использоваться только в соответствии с условиями этого соглашения.

## Авторские права

Авторское право © 1999 г MOXA Technologies Co., Ltd.

Все права сохраняются.

Воспроизведение в любой форме без разрешения запрещено.

## Торговые марки

MOXA - зарегистрированная торговая марка MOXA Technologies Co, Ltd.

Все другие торговые или зарегистрированные марки, упомянутые в настоящем руководстве, принадлежат соответствующим производителям.

## Дополнение

MOXA оставляет за собой право вносить изменения и дополнения в данное руководство без предварительного уведомления потребителя. Не предоставляя гарантий, данное руководство не ограничивает потребителя в решении специфических задач. MOXA оставляет за собой право в любое время изменять и/или модифицировать продукт и/или программное обеспечение, описанные в данном руководстве. MOXA не несет ответственности за использование информации, содержащейся в настоящем руководстве, а также за любые нарушения прав третьих лиц, возникших в результате использования данной информации. Настоящее Руководство может содержать типографские ошибки. Информация, содержащаяся в настоящем руководстве, периодически корректируется; все изменения могут быть включены в новые издания настоящего руководства.

# Об этом руководстве

Это руководство состоит из шести глав и одного приложения. Оно написано для сборщиков, администраторов и системных программистов.

Если Вы - начинающий сборщик или системный администратор, мы рекомендуем Вам изучить издание полностью, за исключением главы 4.

Если Вы - системный программист, ознакомьтесь с главой 4 "Программное обеспечение".

Если Вы нуждаетесь в информации о кабельных соединениях, пожалуйста, см. главу "Соединительные модули (Opt8x) и распайка разъемов".

Если Вы сталкиваетесь с какой-либо проблемой в процессе инсталляции, пожалуйста, обратитесь к главе "Решение проблем".

⇒ В этом руководстве упоминание серии C168 относится к платам C168H и C168HS.

## **Глава 1 Введение**

Эта глава дает краткий обзор и описание возможностей плат серии C168, а также комплект поставки и схему процесса установки.

## **Глава 2 Установка оборудования**

В этой главе предлагается подробное описание установки плат C168 и соединительных модулей (Opt8x).

## **Глава 3 Установка драйвера**

Здесь детально описывается программная инсталляция, настройка, загрузка/выгрузка, обновление и удаление драйвера для операционных систем Windows NT, Windows 95/98, UNIX и DOS.

## **Глава 4 Программное обеспечение**

Эта глава содержит общее описание программного обеспечения под различные операционные системы, включая Pcomm Lite под Windows NT, Windows 95/98, API-232 под DOS и стандартные системные команды UNIX. Также рассматривается вопрос программирования RS-485 (для Opt8J).

## **Глава 5 Соединительные модули (Opt8x) и распайка разъемов**

Эта глава описывает распайки RS-232/422/485 для каждого кабеля (Opt8x).

## **Глава 6 Решение проблем**

В этой главе описываются основные затруднения и возможные варианты их преодоления для серии плат Smartio C168.

## **Приложение Техническая информация**

Здесь дана подробная спецификация, микроконтроллер UART, карта адресов портов ввода-вывода и распайка разъема DB62.

# Оглавление

<b>1. Введение .....</b>	<b>8</b>
Краткий обзор.....	8
Возможности .....	11
Комплект поставки .....	13
Руководство по установке.....	17
<b>2. Установка оборудования .....</b>	<b>18</b>
Настройки по умолчанию .....	18
Быстрая установка.....	19
Как осуществить быструю установку .....	20
Установка с помощью утилиты IO-IRQ.....	21
Утилита IO-IRQ и аппаратная настройка .....	23
<b>3. Установка драйвера .....</b>	<b>29</b>
Windows NT .....	29
Установка драйвера .....	30
Настройка платы и портов .....	37
Обновление драйвера .....	39
Удаление драйвера.....	39
Windows 95/98 .....	40
Установка драйвера. ....	40
Настройка платы и портов .....	46
Обновление драйвера .....	47
Удаление драйвера.....	48
DOS .....	49
Установка драйвера .....	49
Конфигурирование драйвера. ....	50
Загрузка драйвера.....	55
Выгрузка драйвера .....	56
UNIX .....	56
Установка драйвера. ....	57

Назначение имен устройств MOXA TTY .....	61
Настройки скорости .....	62
Административная утилита "мохаadm" .....	62
Подключение терминалов к портам MOXA .....	70
<b>4. Программное обеспечение .....</b>	<b>71</b>
Windows NT and Windows 95/98 .....	71
Инсталляция .....	71
Библиотека функций PCOMM .....	72
Утилиты .....	72
UNIX .....	75
Программирование портов MOXA .....	75
Расширенные команды UNIX ioctl () .....	76
Утилиты .....	82
DOS .....	85
Инсталляция .....	85
Библиотека DOS API-232 .....	85
Утилиты .....	86
Программирование RS-485 для Opt8J .....	88
<b>5. Соединительные модули (Opt8x) и распайка разъемов ..</b>	<b>91</b>
Распайка разъемов RS-232 Opt8A/B/C/D/S .....	91
Распайка разъемов RS-422 для Opt8J/F/Z .....	97
Распайка разъема RS-485 для Opt8J .....	99
RS-422/485 Согласование импедансов .....	100
<b>6. Решение проблем .....</b>	<b>102</b>
Решение общих проблем .....	102
Windows NT .....	106
Windows 95/98 .....	107
DOS .....	108
UNIX .....	108
<b>Приложение. Техническая информация .....</b>	<b>110</b>
Спецификация .....	110

UART 16C550C.....	111
Карта адресов ввода-вывода портов.....	112
Контакты разъема DB62.....	113

# 1. Введение

## Краткий обзор

### Мощная мультипортовая асинхронная система Smartio

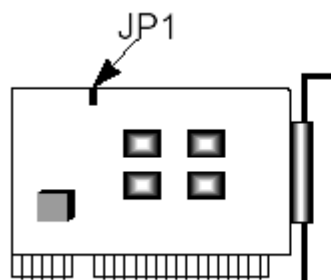
Термин Smartio применяется к мощным мультипортовым системам ввода-вывода. Серия плат **Smartio C168** поддерживает 8 последовательных портов, предназначенных для связи с терминалами, модемами, принтерами, устройствами сбора данных и другими последовательными устройствами для PC и других PC/AT-совместимых систем. Благодаря обширным настройкам драйвера, платы Smartio могут полностью использовать 32-битный Tx/Rx FIFO и встроенный в микросхему аппаратный контроль потока данных, что позволяет без потерь осуществлять передачу данных со скоростью до 921.6 Kbps. Таким образом, предоставляется возможность для налаживания надежной и высокопроизводительной последовательной мультипортовой связи.

Платы серии Smartio C168 оборудованы **специально разработанным кристаллом ASIC**, который заменяет большое количество обычных интегральных микросхем и, следовательно, приводит к уменьшению платы до половинного размера. Данная серия поддерживает 16-битную архитектуру. Также является доступным весь диапазон адресов ввода-вывода и IRQ. Кроме этого, в плату встроена система **EEPROM**, сохраняющая данные конфигурации, поэтому вся серия не имеет перемычек и переключателей. Это позволяет сделать каждый порт на плате действительно независимым от других портов и добиться совместимости с большинством имеющихся мультипортовых плат.

Существует также разработка платы серии Smartio C168 для PCI слота. Для получения более подробной информации обратитесь к дилеру/дистрибьютору или на веб-сайт MOXA.

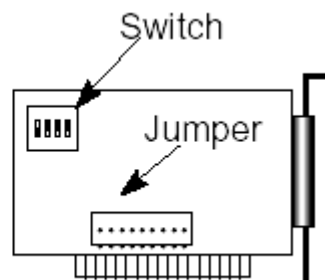
## Методика настройки платы

Плата серии Smartio



**Новое:** установите адреса ввода-вывода и IRQ с помощью программной утилиты.

Обычная мультипортовая плата



**Традиционное:** установите переключатель и перемычку вручную для адреса и IRQ.

Вместо перемычек и переключателей на обычных платах, используемых для выставления значений IRQ и адреса ввода-вывода, настройка каждого порта осуществляется с помощью утилиты под DOS, **Io-irq.exe**, которая считывает и записывает информацию о настройках во встроенный EEPROM через адрес **САР (Порт доступа к конфигурации)**. САР адрес – это канал, через который конфигурационная утилита Io-irq.exe осуществляет доступ к настройкам платы и идентичен **адресу ввода-вывода первого порта**.

Единственная имеющаяся перемычка **JP1** предназначена на случай, если пользователь забыл САР адрес. В обычной ситуации перемычка всегда открыта. Если эту перемычку закрыть, то САР адрес принудительно принимает изначально установленное значение адреса ввода-вывода, **0xA700**. Например, чтобы осуществить



быструю инсталляцию (см. ниже), перемычка должна находиться в закрытом состоянии.

## Быстрая инсталляция

Чтобы упростить настройку платы, пользователю, устанавливающему **только одну плату Smartio C168 под системы Windows NT/95/98**, рекомендуется использовать **быструю инсталляцию**, описанную в главе 2.

Благодаря очень гибким параметрам настройки платы данной серии совместимы с большинством производимых мультипортовых плат, использующих UART 16450 или 16550.

## Защита от выбросов напряжения

Чтобы защитить платы от повреждений, связанных с ударом молнии или попаданием высокого потенциала, некоторые модификации плат созданы с использованием **технологии защиты от выбросов напряжения**.

## Поддержка операционных систем

Платы серии Smartio C168 поддерживаются наиболее популярными операционными системами, такими как Windows NT, Windows 95/98, SCO UNIX/XENIX/OpenServer, DOS, Linux, QNX, FreeBSD, UNIX SVR4.2. Например, **драйвер устройства MOXA** для Windows NT, Windows 95/98, Linux, SCO UNIX/OpenServer, SVR4.2 и DOS обеспечивают простую инсталляцию, настройку и отличную производительность. В этом руководстве содержатся описания установки драйвера под **Windows NT, Windows 95/98, UNIX и DOS**. Другие совместимые системы здесь не упоминаются; обратитесь к соответствующему системному руководству, где будет описана процедура установки и настройки стандартного драйвера.

## Мощная программная поддержка при работе с последовательными портами

Для удобства пользователя, MOXA предоставляет простые в использовании коммуникационные библиотеки под Windows NT/95/98 (**PCComm Lite**) и DOS (**API -232**). Пользователи могут использовать эти библиотеки для того, чтобы разработать свои собственные приложения на Microsoft C, Turbo C, Assembler, QuickBasic, Turbo Pascal, Clipper, Visual Basic, Visual C++, Borland Delphi и т.д. **Утилиты**, такие как Diagnostic и Monitor, используются для проверки платы/порта или для наблюдения за состоянием линии связи.

## Сферы применения

Smartio C168 применима во многих областях деятельности.

Некоторые из них приведены ниже:

- ❖ Internet / локальные сети
- ❖ Работа с удаленным доступом
- ❖ Многопользовательские системы
- ❖ Автоматизация производства
- ❖ Офисная автоматизация
- ❖ Телекоммуникации
- ❖ Торговые автоматы на базе PC
- ❖ Система учета товаров.

## Возможности

Серия плат Smartio C168 имеет следующие модификации:

**C168H:** 8-портовая плата RS-232 или RS422, UART 16550C или совместимые

**C168HS:** 8-портовая плата RS-232 или RS422, UART 16550C или совместимые, защита от выбросов напряжения.

- Высокоинтегрированная микросхема ASIC Компактный размер платы
- Отсутствие переключателей и перемычек, простая программная настройка
- Независимые адреса ввода-вывода и значения IRQ для каждого из восьми последовательных портов
- 16-битная архитектура, поддерживающая большее количество значений IRQ
- Защита от выбросов напряжения для RS-232 (C168HS)
- Защита от перепадов напряжения для RS-422 (с помощью соединительного кабеля Opt8F)
- Высокоскоростной контроллер связи 16C550C с аппаратным контролем потока, не допускающий потерю данных
- Поддержка популярных операционных систем: Windows NT/95/98, SCO UNIX/OpenServer, UNIX SVR4.2, DOS, Linux
- Совместимость с многими другими операционными системами: QNX, SCO XENIX, Free BSD

C168H/HS

Windows NT	+
Windows 95/98	+
DOS	+
SCO UNIX/OpenServer	+
UNIX SVR4.2	+
Linux	R
QNX	C
SCO XENIX	C
Free BSD	C

+ : драйвер MOXA поставляется с изделием

R : драйвер MOXA поставляется после запроса

C: драйвер поддерживается операционной системой

Обратите внимание: вы всегда можете скачать последнюю версию драйвера с FTP сервера MOXA

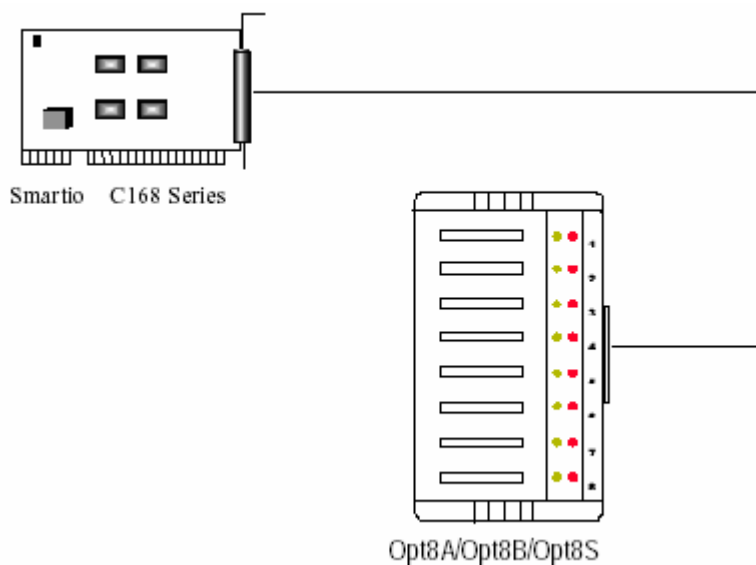
## Комплект поставки

- 8-портовая плата серии Smartio C168
- Компакт-диск с программным обеспечением MOXA, на котором присутствуют драйвера:
  - Windows NT и Windows 95/98
  - DOS
  - UNIX
- Руководство пользователя Smartio C168H/HS
- Пакет программ PComm Lite

Вам также потребуется один из перечисленных ниже соединительных модулей:

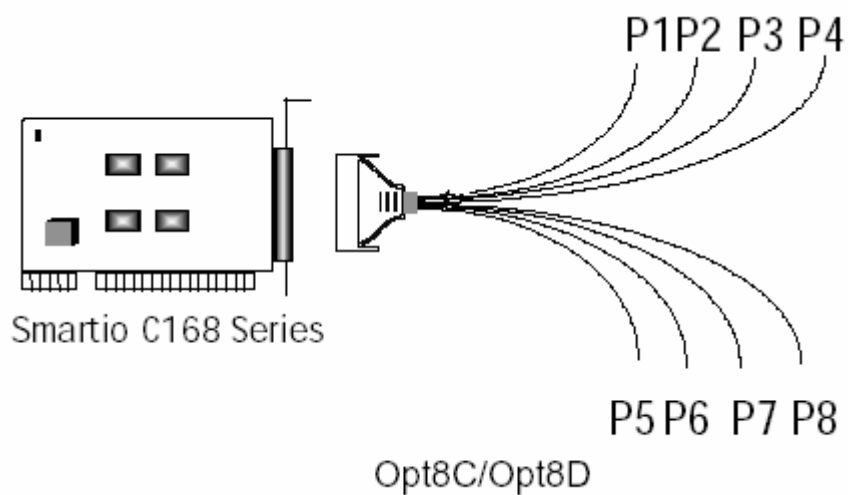
### Opt8A/Opt8B/Opt8S

- ❖ RS-232 соединительный модуль с 8 DB25 (гнездо/штекер/гнездо) разъемами (защита от выбросов для Opt8S).
- ❖ Кабель 1.5 метра DB62 - DB62.



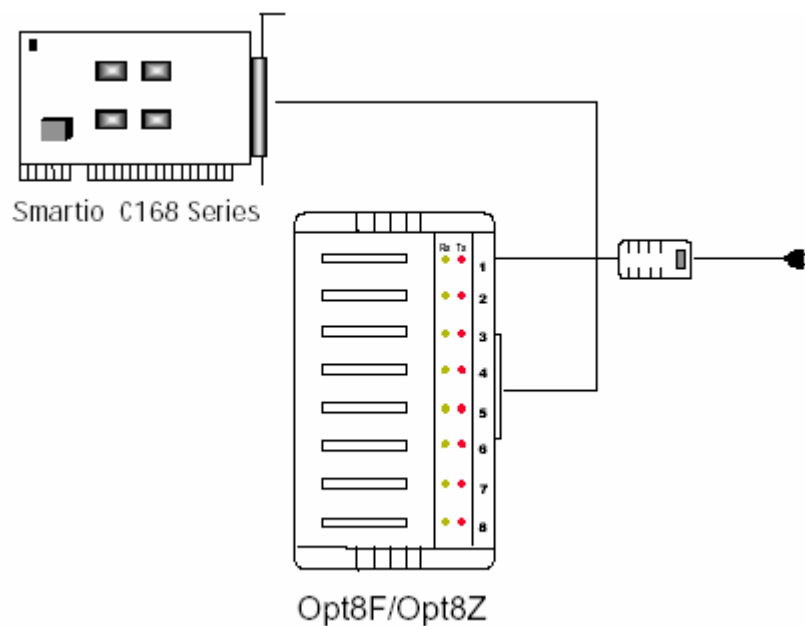
## Opt8C/Opt8D

- ❖ RS-232 кабель с 8 разъемами DB25 (штекер) для Opt8C и DB9 для Opt8D (длина 1 метр).



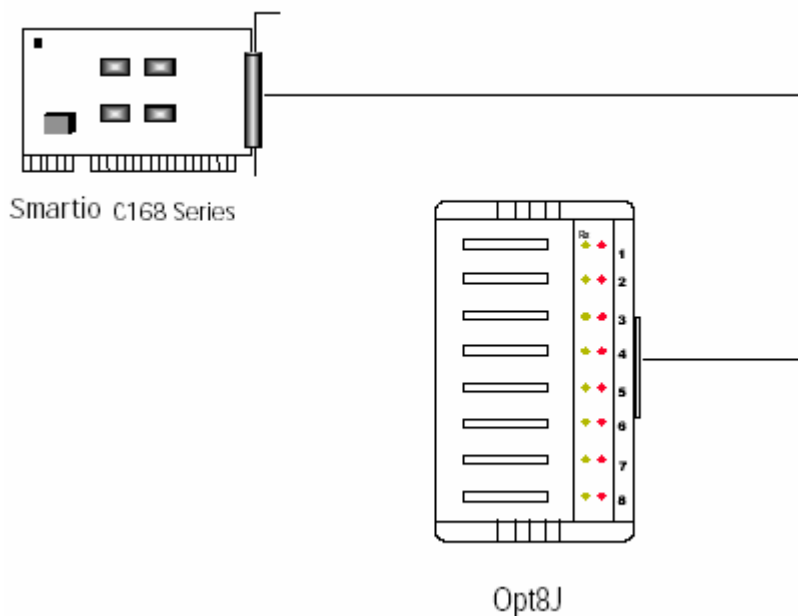
## Opt8F/Opt8Z

- ❖ RS-422 соединительный модуль с 8 разъемами DB25 (гнездо) (защита от выбросов для Opt8F).
- ❖ 1.5 метра кабель DB62 - DB62.
- ❖ Адаптер 110V или 220V.

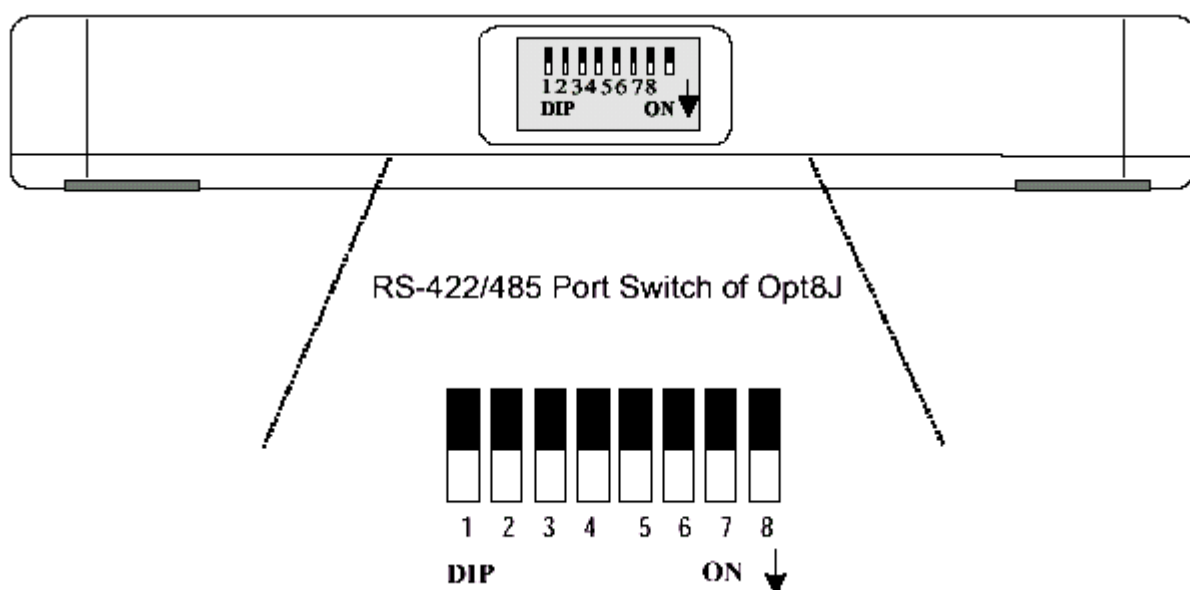


## Opt8J

- ❖ RS-422/485 соединительный модуль с 8 разъемами DB25 (гнездо).
- ❖ 1.5 метра кабель DB62 - DB62.
- ❖ Адаптер 110V или 220V.



**Opt8J** - это соединительный модуль RS-422/485 с 8 разъемами DB25 (гнездо) для восьмипортовых плат MOXA, включая Smartio C168. На Opt8J имеются 8 DIP-переключателей. Каждый переключатель управляет режимом связи (RS-422 или RS-485) соответствующего порта.



## Режим RS-422

Чтобы запустить работу с интерфейсом RS-422, установите соответствующий переключатель в позицию **OFF**. Это означает что порт всегда готов к одновременной передаче и принятию данных, т. е. к работе в дуплексном режиме.

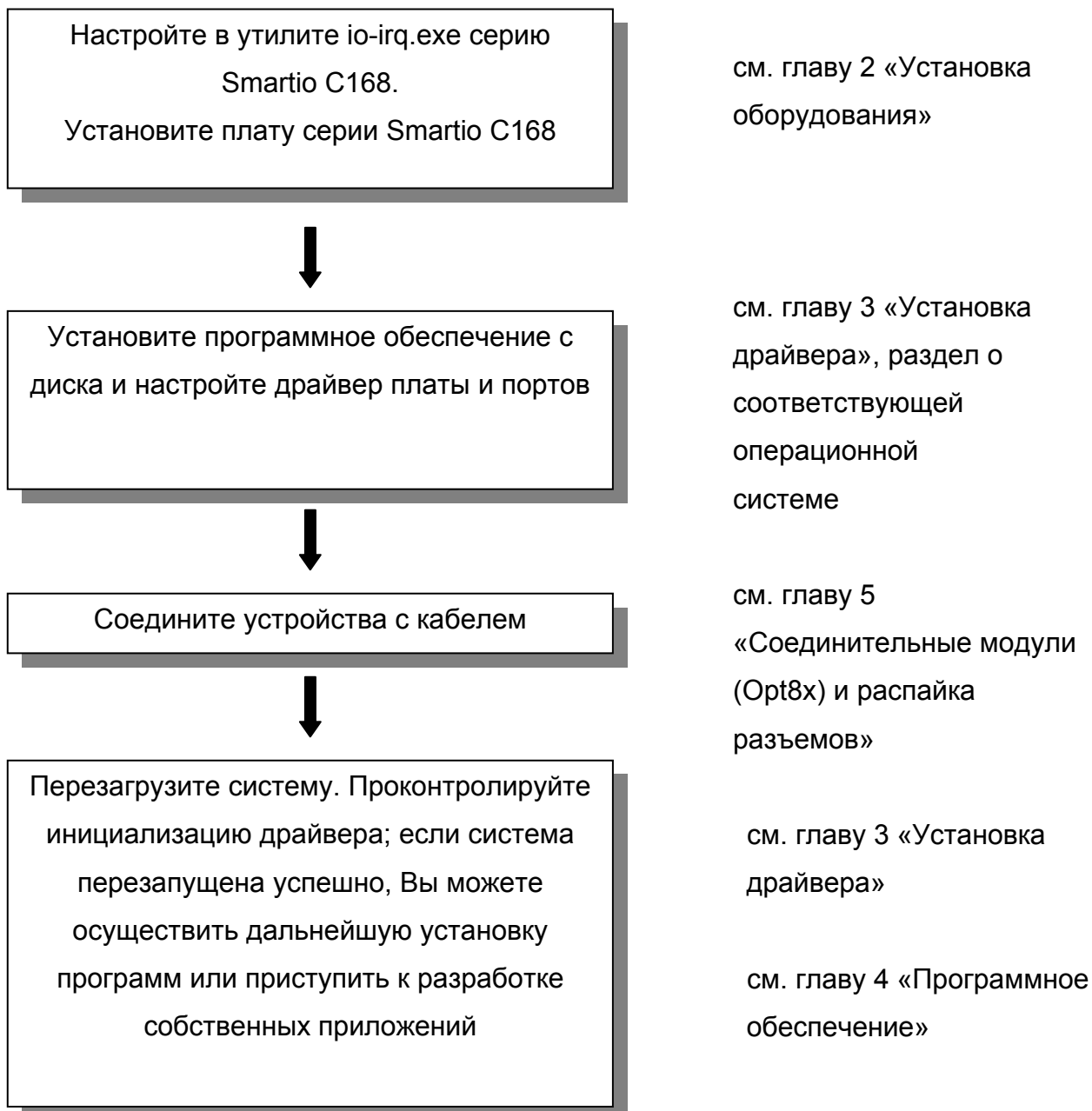
## Режим RS-485

Opt8J поддерживает только 2-проводную RS-485 связь. Чтобы запустить работу с интерфейсом RS-485, установите соответствующий переключатель в позицию **ON**. Это означает, что порт может передавать данные только тогда, когда установлен сигнал RTS, и принимать данные только когда сигнал RTS выключен (полудуплекс).

Для подключения кабелей RS-422/RS-485 обратитесь к главе “Соединительные модули (Opt8x) и распайка разъемов”, а также к главе “Программное обеспечение” для ознакомления с нюансами программирования Opt8J RS-485.

# Руководство по установке

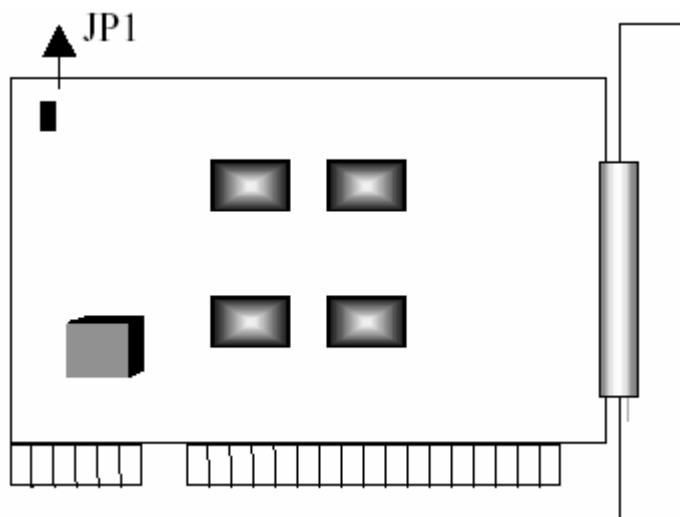
Этот раздел дает краткое описание процесса установки платы серии Smartio C168. Инсталляция проста и включает следующие стадии:





## 2. Установка оборудования

Процесс установки платы серии Smartio C168 состоит из установки оборудования и инсталляции программного обеспечения. Здесь будет рассматриваться установка платы в компьютер. Инсталляция драйверов под различные операционные системы подробно обсуждается в следующей главе.



Серия Smartio C168

### Настройки по умолчанию

По умолчанию платы серии Smartio C168 имеют следующие установки:

Адреса ввода-вывода: **0x180**(порт 1), **0x188**(порт 2), **0x190**(порт 3),  
**0x198**(порт 4), **0x1A0**(порт 5), **0x1A8**(порт 6),  
**0x1B0**(порт 7), **0x1B8**(порт 8)

IRQ: **10**

Вектор прерывания: **0x1C0**

Переключатель CAP JP1: **Открыта**

**Обратите внимание:** если установки по умолчанию соответствуют вашим желаниям и позволяют системе работать без конфликтов, тогда просто установите плату в систему и переходите непосредственно к следующей главе «Установка драйвера». В обратном случае, следуйте инструкциям, описанным ниже.

Теперь вы можете осуществить **обычную установку** (подробности описаны в следующем разделе «Установка с помощью утилиты IO-IRQ») или **быструю установку** (см. раздел «Быстрая установка»). При быстрой установке имеются следующие нюансы:

- в систему может быть установлена **только одна плата серии Smartio C168**
- поддерживаются только операционные системы **Windows NT и 95/98**
- адрес ввода-вывода **0xA700** должен быть свободен

## Быстрая установка

Быстрый и простой метод установки платы Smartio C168 **без использования утилиты настройки Io-Irq.exe** разработан для пользователей, которые совершенно не нуждаются в специфичных аппаратных настройках; он позволяет плате использовать весь спектр поддерживаемых возможностей и гибких настроек. В данном случае перемычка JP1 всегда должна быть закрыта. Аппаратная и программная настройка будут завершены одновременно в процессе настройки драйвера.

Соответственно, по умолчанию устанавливается значение скорости в интервале от 50 до 921.6 K bps, который называется

**высокоскоростной режим (High Speed Spectrum)** и подробно описывается в разделе «Установка с помощью утилиты IO-IRQ».

## Как осуществить быструю установку

Пользователям, которые устанавливают **только одну плату серии Smartio C168 под Windows NT/95/98**, настоятельно рекомендуется использовать следующий метод быстрой установки:

1. **Закройте переключку JP1**, расположенную в верхнем левом углу платы.
2. Вставьте плату в предварительно отключенный от сети компьютер.
3. Осуществите установку драйвера, подробно описанную в следующей главе.

Это позволит установить в панели конфигурации желаемые адреса ввода-вывода, IRQ и вектора прерывания, не зависящих от аппаратных настроек платы. Программная настройка автоматически изменит их.

После этого установка будет полностью завершена.

4. Закройте систему (Windows NT/95/98).
5. Полностью отключите питание (OFF), а затем включите его (ON).  
**(Осуществите «холодный запуск»)**
6. Перезапустите систему (Windows NT/95/98).

В данном случае очень важно, переключка JP1 всегда была закрыта. Программа настройки автоматически изменит аппаратные установки платы на заданные в программных настройках без запуска утилиты

io-irq.exe под DOS. Это устраняет возможные сложности при осуществлении аппаратной настройки. Тем не менее необходимо помнить, что при любых изменениях в конфигурации **необходимо осуществлять «холодный запуск» системы.**

## Установка с помощью утилиты IO-IRQ

Этот раздел следует изучить тем пользователям, которые не будут использовать быструю установку по причине:

- Установки в систему двух и более плат Smartio C168.
- Невозможности установить адрес ввода-вывода 0xA700 или конфликта в системе.
- Использования иной операционной системы, кроме Windows NT/95/98.

Перед **установкой драйвера**, детально описанной в следующей главе «Установка драйвера», осуществите **установку оборудования**, позволяющую настроить адреса ввода-вывода и IRQ в утилите "**io-irq.exe**". Подробности изложены в следующем разделе.

По умолчанию платы серии Smartio C168 имеют следующие настройки:

Адреса ввода-вывода:	<b>0x180</b> (порт 1), <b>0x188</b> (порт 2), <b>0x190</b> (порт 3), <b>0x198</b> (порт 4), <b>0x1A0</b> (порт 5), <b>0x1A8</b> (порт 6), <b>0x1B0</b> (порт 7), <b>0x1B8</b> (порт 8)
IRQ:	<b>10</b>
INT Vector:	<b>0x1C0</b>

Так как платы серии Smartio C168 не имеют переключателей и переключателей, то для ручной установки адресов ввода-вывода, IRQ и INT Vector вы **обязательно** должны с диска запустить

программу настройки **Io-irq.exe** под операционной системой **DOS**, чтобы изменить аппаратные настройки платы.

1. Запустите на персональном компьютере операционную систему **DOS**.
2. Выключите питание.
3. Убедитесь в отсутствии аппаратной несовместимости и установите плату в 16-битную шину на материнской плате, **только одну плату с открытой перемычкой JP1**.

- Если вам нужно установить несколько плат, вставьте одну плату и настройте ее с использованием утилиты **Io-irq.exe** перед установкой следующей. Это поможет избежать конфликта некоторых аппаратных установок между двумя платами.

По умолчанию платы серии Smartio C168 имеют следующие настройки:

Адреса ввода-вывода: **0x180**(порт 1), **0x188**(порт 2),  
**0x190**(порт 3), **0x198**(порт 4),  
**0x1A0**(порт 5), **0x1A8**(порт 6),  
**0x1B0**(порт 7), **0x1B8**(порт 8)

IRQ: **10**

INT Vector: **0x1C0**

CAP: **0x180**

4. Включите компьютер и войдите в систему DOS.
5. Запустите утилиту **Io-irq.exe** с диска с драйверами, чтобы настроить адреса ввода-вывода, IRQ и вектор прерывания.

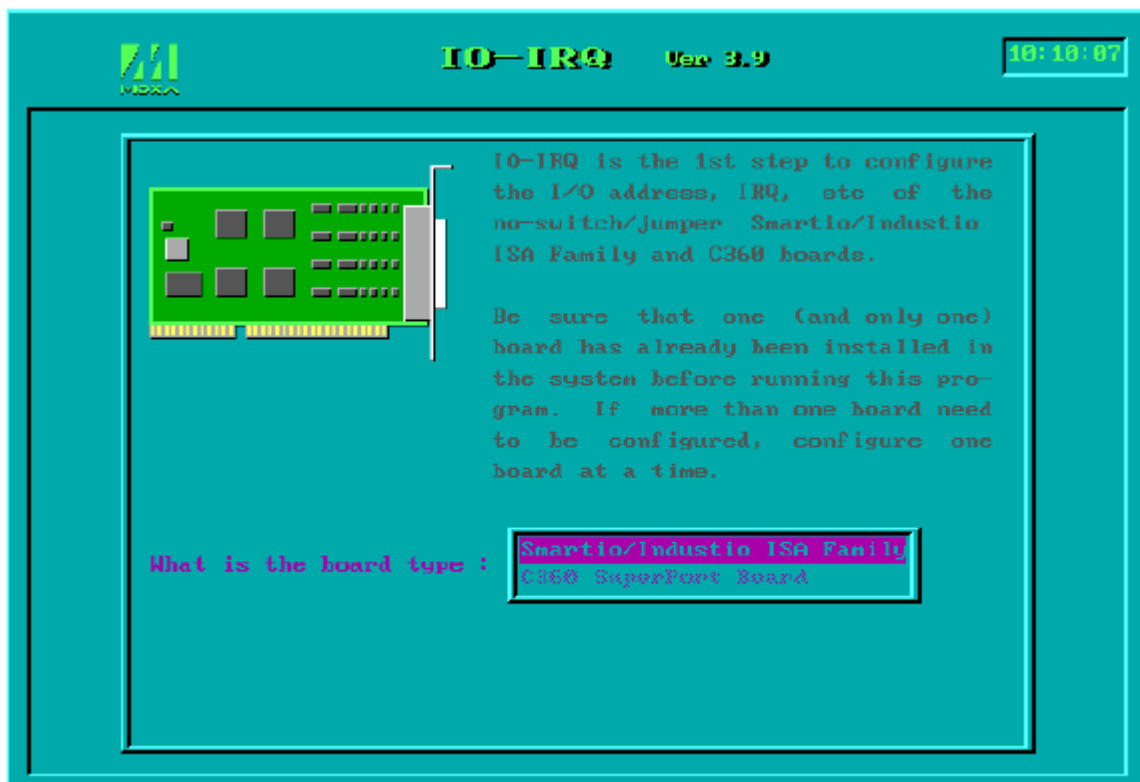
После того, как будут завершена аппаратная настройка платы, то она будет готова под использование под различными

операционными системами, такими как Windows NT и 95/98, DOS, UNIX и пр.

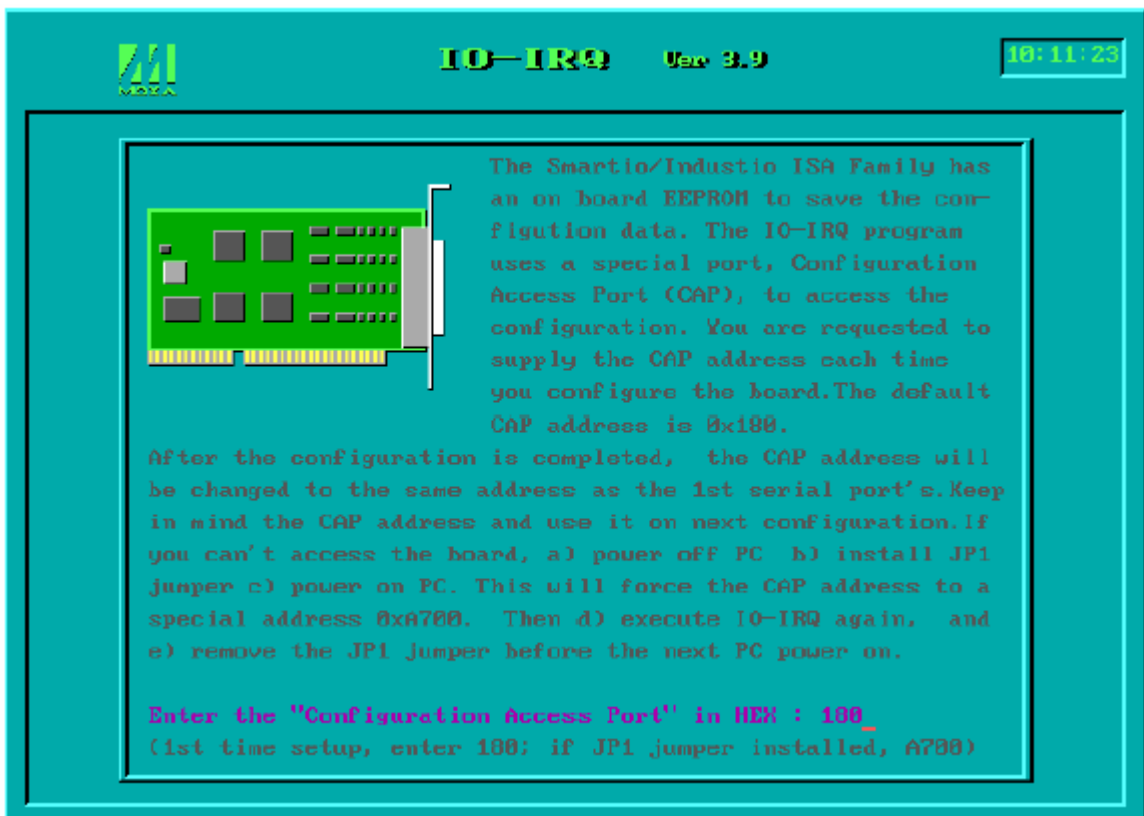
## Утилита IO-IRQ и аппаратная настройка

Обратите внимание на то, что значение CAP адреса, 0x180, соответствует **адресу ввода-вывода первого порта**, за исключением тех случаев, когда перед включением компьютера была установлена перемычка JP1. Тогда адрес CAP принимает значение **0xA700**. Адрес CAP может быть изменен. Чтобы его скорректировать, утилита считывает его из **EEPROM** (интегрированной в плату) и отображает в конфигурационной панели. CAP адрес— это канал доступа, через который утилита io-irq.exe изменяет настройки платы.

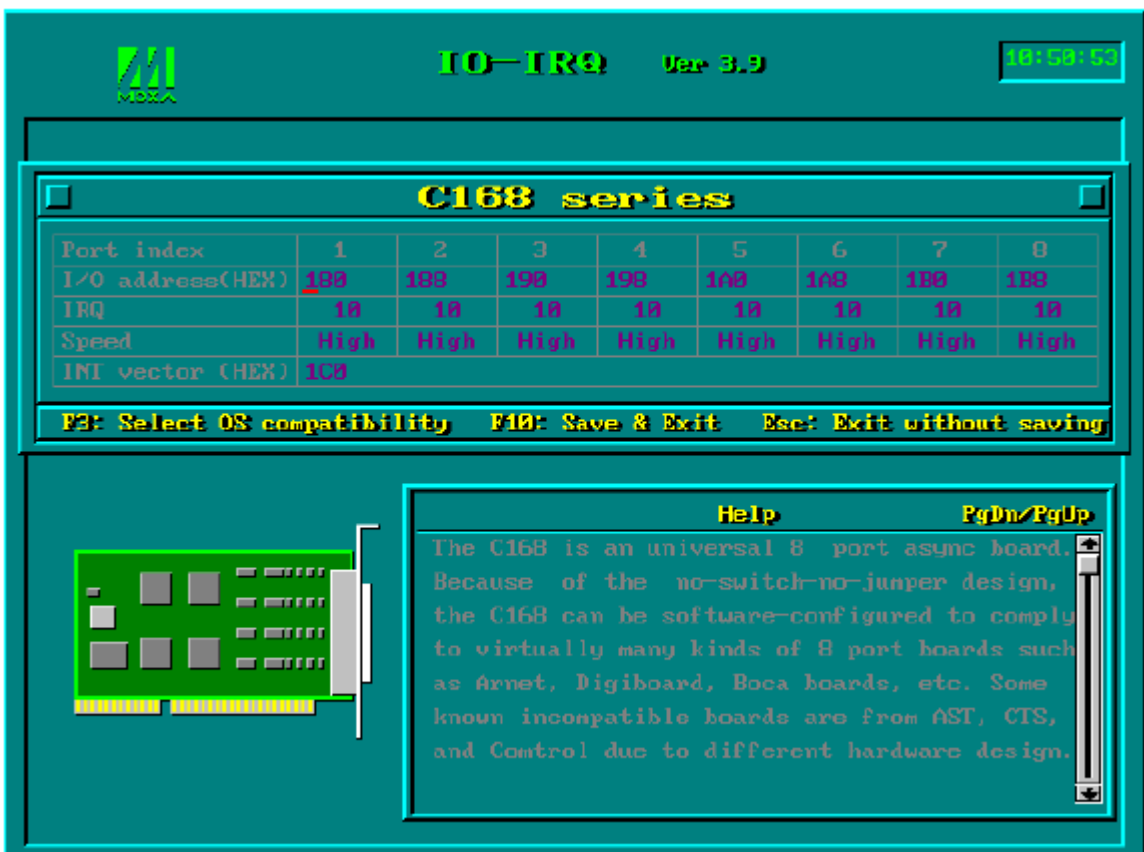
1. Запустите утилиту **io-irq.exe** с диска с драйверами для установки на плате адресов ввода-вывода, IRQ и вектора прерывания.



2. Выберите «Smartio/Industio ISA Family» и нажмите кнопку ENTER.

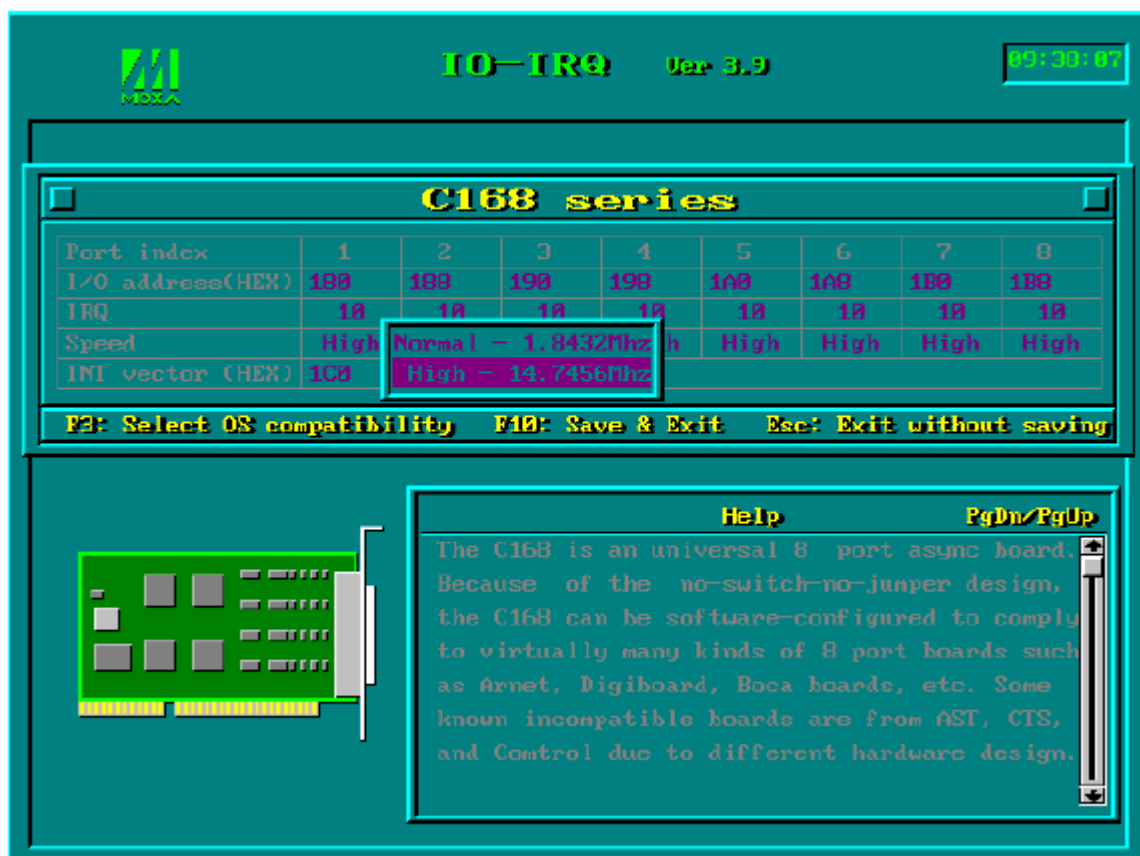


3. Введите CAP адрес платы серии Smartio C168, чтобы изменить его.



4. При необходимости настройте следующие параметры.

- Port Index** Укажите номер каждого порта.
- I/O address** Введите адреса ввода-вывода для каждого порта, последовательно или нет. Избегайте конфликтов с другими устройствами.
- IRQ** Введите IRQ для каждого порта (2, 3, 4, 5, 7, 10, 11, 12, 15).
- Speed** Это поле определяет использование **нормального** или **высокоскоростного режимов**. Нормальная скорость определяется в интервале от 50 bps до 115.2 Kbps, высокая – от 50 bps до 921.6 Kbps. Платы серии Smartio C168 поддерживают оба спектра скоростей.



Обратите внимание на то, что порты, использующие **драйвера MOXA под Windows NT и Windows 95/98** будут отражать реальную



рабочую скорость и в **нормальном**, и в **высокоскоростном** режимах. Например, отображаемая скорость 38.4 Kbps в действительности будет соответствовать реальной скорости.

При этом если порты используют **драйвера не-МОХА**, такие как **стандартные последовательные драйвера**, или драйвер МОХА под другую операционную систему, например, **DOS**, реальная рабочая скорость **в восемь раз** превышает отображаемую. Например, порт, отображающий в нормальном режиме (Normal Speed Spectrum) 38.4 Kbps, в действительности и будет работать на этой скорости. Если же он работает в высокоскоростном режиме (High Speed Spectrum) и показывает 38.4 Kbps, то он будет работать с реальной скоростью 307.2 Kbps (т.е. 38.4 Kbps x 8).

Ниже приводится таблица **соответствия скоростей** для быстрой ориентации, в частности для драйвера DOS.

<u>Normal Speed Spectrum</u>	<u>High Speed Spectrum</u>
50 (bps)	400 (bps)
75	600
110	880
134.5	1076
150	1200
300	2400
600	4800
1200	9600
1800	14.4K
2400	19.2K

4800	38.4K
7200	57.6K
9600	76.8K
19.2K	153.6K
38.4K	307.2K
57.6K	460.8K
115.2K	921.6K

### **INT Vector**

Введите адрес вектора прерывания. Значение этого адреса для вектора прерывания лежит в интервале от 00000H до 0FFFFH. Под вектором прерывания, в данном случае, подразумевается специальный регистр в области ввода-вывода, в котором каждый бит используется для отображения приходящего с порта прерывания. Номер бита в данном регистре соответствует номеру порта с которого приходит прерывание. Чтобы использовать вектор прерывания, установите в аппаратных настройках, для него, адрес лежащий в указанном выше интервале. Если же вектор прерывания не будет использоваться, оставьте его пустым.

Есть два режима работы драйвера плат серии Smartio C168. Первый использует вектор прерывания, второй – нет. Использование драйвером вектора прерывания позволяет добиться более высокой производительности работы платы.

5. Нажмите кнопку **F10** чтобы сохранить настройки и выйдите из утилиты.

```
The "Configuration Access Port" for  
this board is : 0x180  
Use this address to configure the  
board next time, and press any key  
to exit.
```

## 3. Установка драйвера

В этом разделе описываются процедуры установки, настройки, обновления/удаления драйвера в различных операционных системах: Windows NT, Windows 95/98, UNIX и DOS. Прежде, чем приступить к программной инсталляции, должна быть осуществлена установка платы, детально описанная в предыдущей главе «Установка оборудования».

Если вы хотите разрабатывать собственные приложения, пожалуйста, прочтите следующую главу «**Программное обеспечение**» для ознакомления с нюансами программирования последовательного интерфейса.

### Windows NT

Windows NT поддерживает до **256** последовательных портов, от **COM1** до **COM256**. Чтобы полностью реализовать интегрированные возможности Windows NT, для плат серии Smartio C168 и других мультипортовых плат разработаны специальные 32-разрядные драйверы. Драйвер соответствует стандарту Win32 API COM.

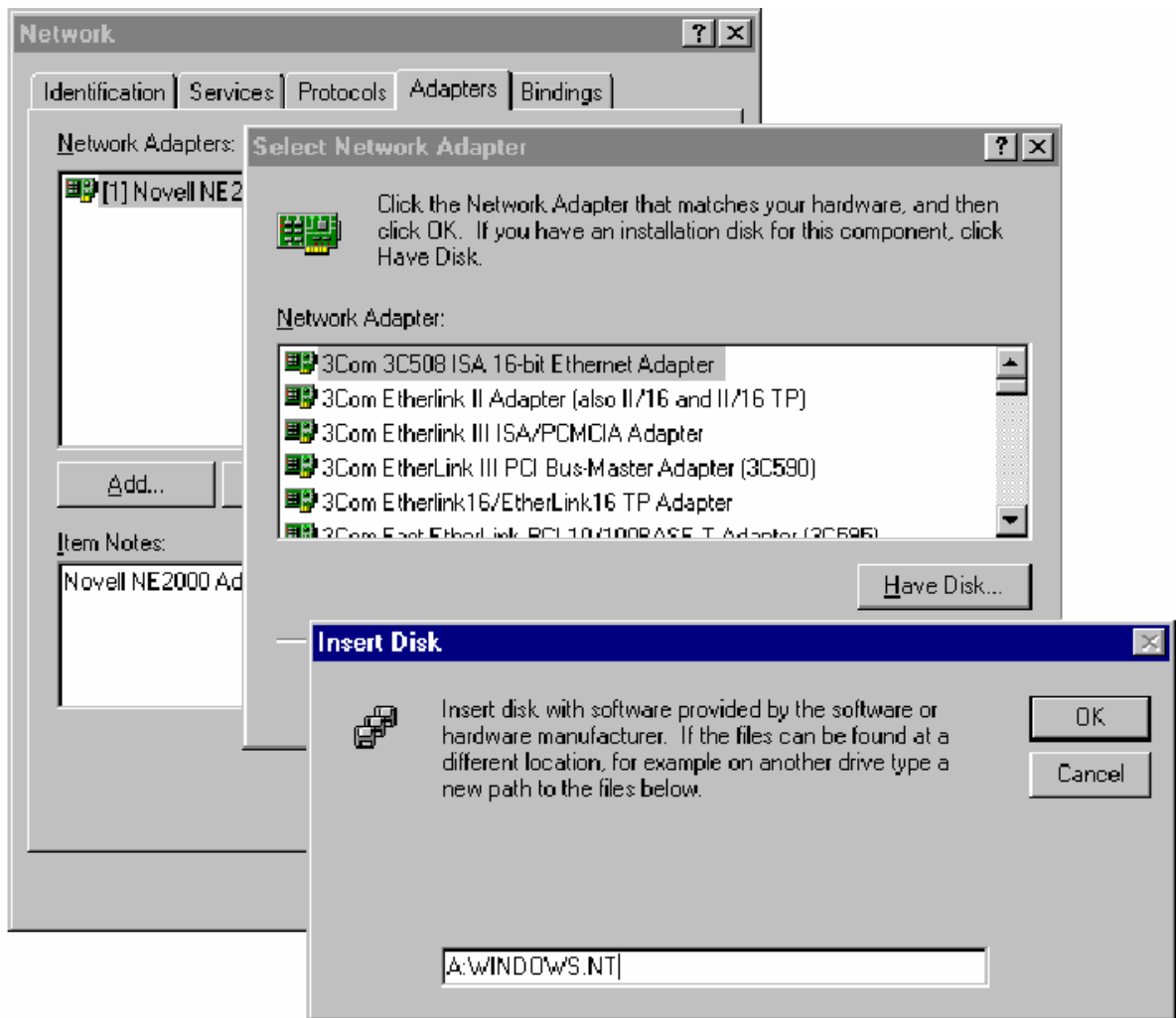
- Если Вы впервые устанавливаете драйвер, тогда, пожалуйста, обращайтесь непосредственно к разделу "Установка драйвера".
- Если хотите перенастроить драйвер и порты установленной платы добавить новые или удалить имеющиеся платы, обратитесь к разделу "Настройка платы и портов".
- Если Вы хотите обновить или удалить драйвер, прочтите раздел "Обновление драйвера" или "Удаление драйвера".

## Установка драйвера

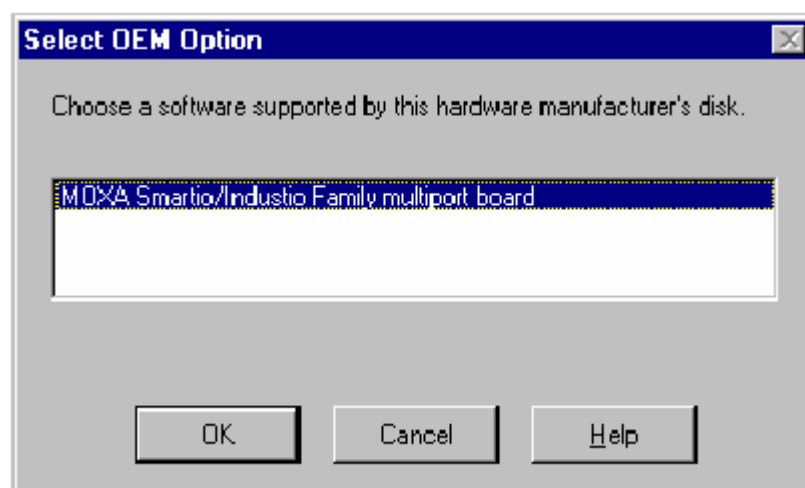
Далее рассматривается процедура **первичной** установки драйвера платы серии Smartio C168 под систему Windows NT 4.0.

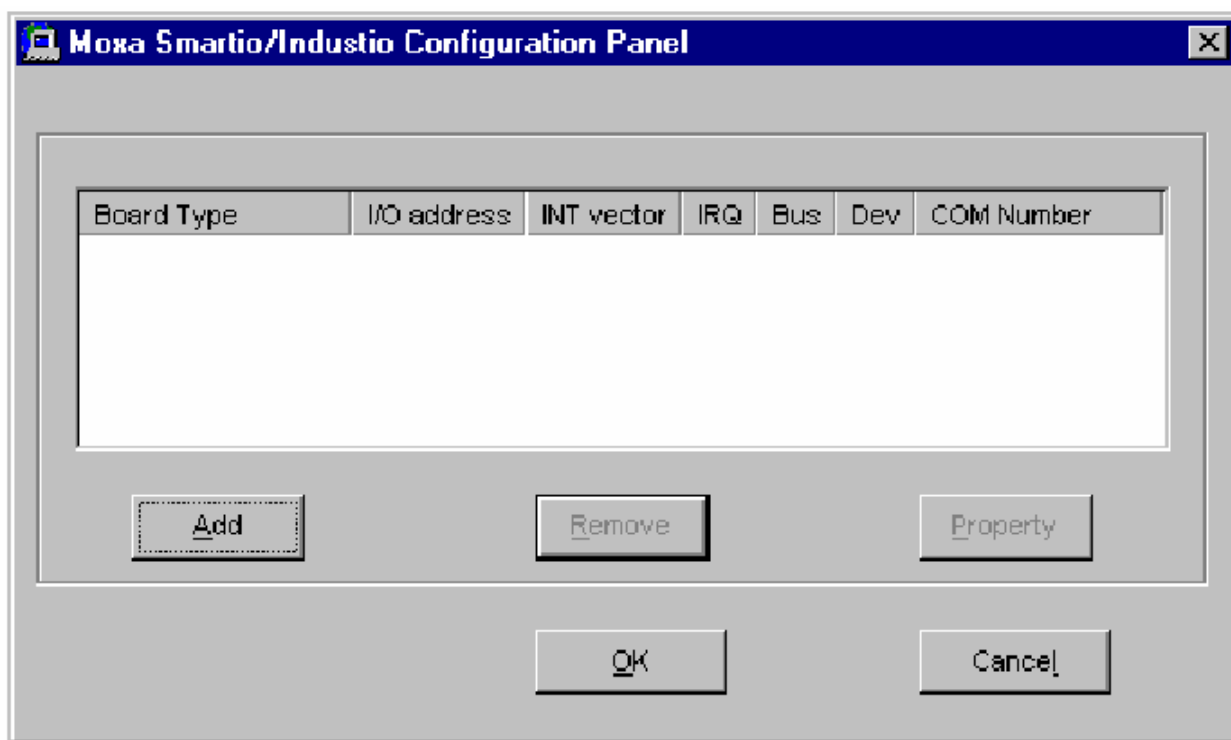
**Обратите внимание:** убедитесь, что плата(ы) уже установлены в систему, если вы осуществляете **быструю установку**.

1. Пожалуйста, войдите в систему как **Администратор**.
2. Выберите **Control Panel** [Панель управления], откройте значок **Network** [Сеть] и затем выберите закладку **Adapters** [Устройства].
3. Нажмите кнопку **Add** [Добавить], затем кнопку **Have Disk** [Установить с диска...] в окне Select Network Adapter [Выбор сетевой платы].
4. Укажите путь к диску с драйвером, **X:WINDOWS.NT**. Затем нажмите на кнопку **[OK]**.

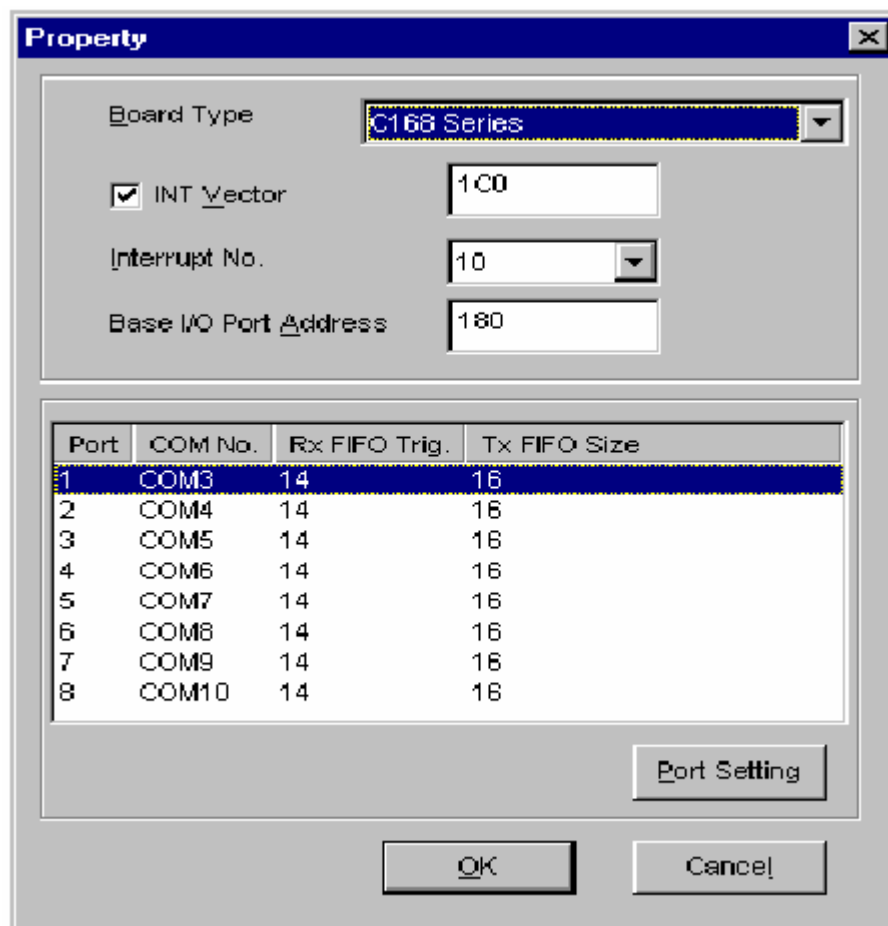


5. Выберите пункт «**MOXA Smartio/Industio Family multiport board**» в диалоговом окне Select OEM Options и щелкните кнопку [OK], чтобы вызвать открыть диалоговое окно «**MOXA Smartio/Industio Family Configuration Panel**» и запустить начало установки драйвера.





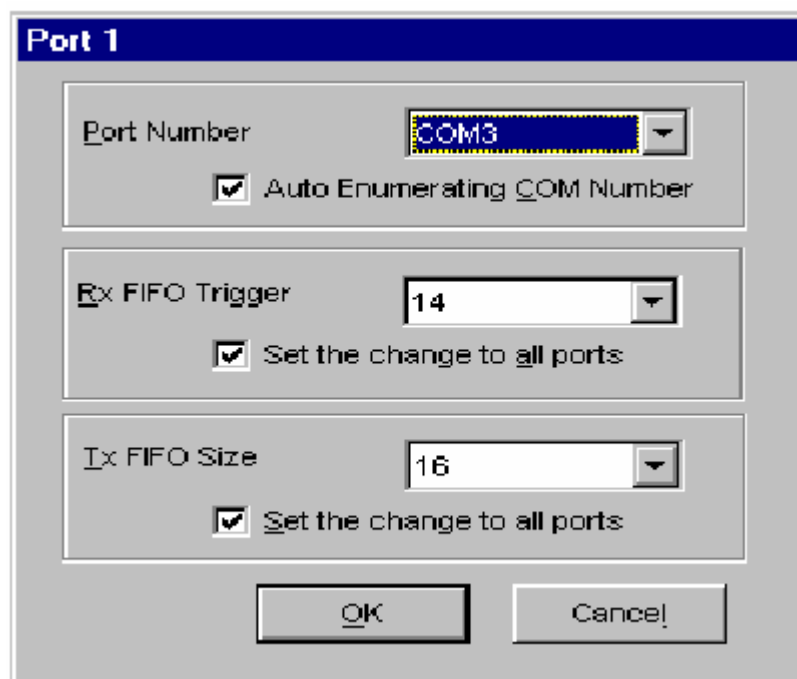
6. В диалоге «**Moxa Smartio/Industio Configuration Panel**» нажмите на кнопку **Add**, открывая диалог **Property**, чтобы добавить новую плату серии Smartio C168. В поле «Board Type» выберите пункт «**C168 Series**». При необходимости, установите желаемое значение вектора прерывания в поле «INT Vector». Выберите нужный номер прерывания в поле «Interrupt No» и номер адреса ввода-вывода в поле «Base I/O Port Address». Все эти настройки должны совпадать с аппаратными настройками платы и не конфликтовать с другими устройствами.



**Обратите внимание:** если у вас нет необходимости изменять другие какие-либо настройки, то переходите к **шагу 8**.

7. В диалоге «**Property**» выберите из списка желаемый порт и нажмите кнопку [**Port Setting**], чтобы открыть диалоговое окно «**Port #**», позволяющем изменить COM номера или настройки FIFO.





- **Номер порта**

Вы должны установить все порты платы с желаемыми номерами **COM**, которые не должны противоречить другим занятым номерам COM. В этом окне есть два пути придания физическим портам соответственных COM номеров в зависимости от переключателя "**Auto Enumerating COM number**".

Если установлен флажок "Auto Enumerating COM number", то определите номер первого порта, то остальным будут присвоены последовательно возрастающие номера COM. Например, если первый порт отображен как COM3, то второй порт отображается соответственно как COM4.

Если флажок "Auto Enumerating COM number" не установлен, то вы должны определить номер для каждого COM порта. Например, второй порт может быть обозначен как COM10, в то время как первый порт отображается как COM3.

- **Rx FIFO Trigger**

Rx FIFO может принимать следующие значения: 1, 4, 8 или 14 байт; по умолчанию задается значение 14 байт.

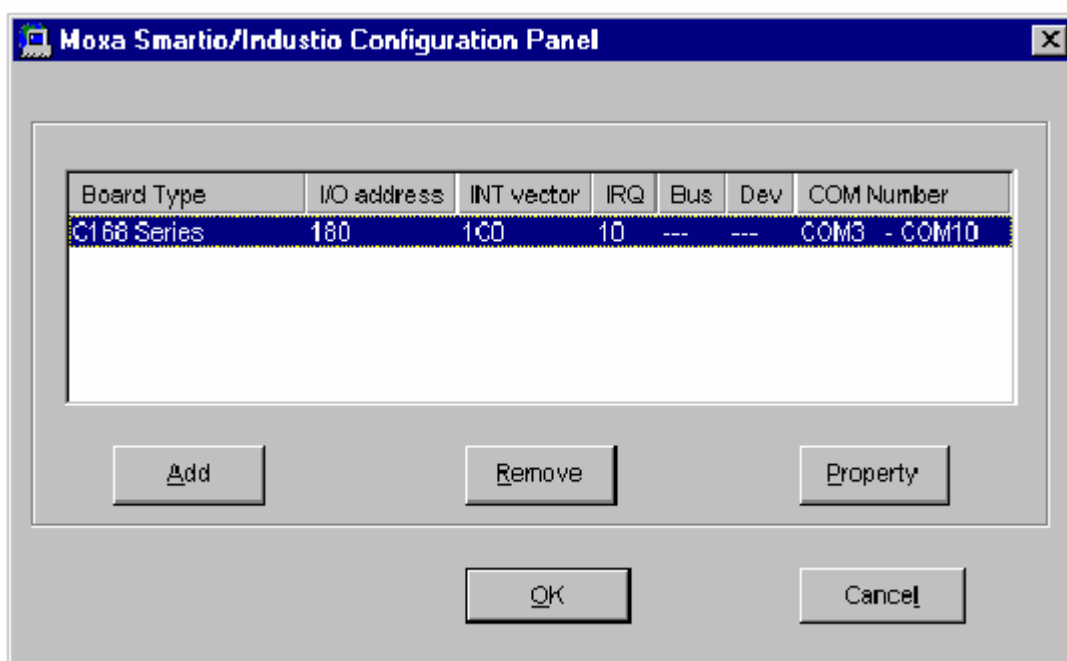
- **Tx FIFO Size**

Допустимое значение Tx FIFO – от 1 до 16 байт. По умолчанию задается 16 байт.

8. Нажмите кнопку **[OK]** в диалогах «Port #» и «Property», чтобы вернуться к окну «Moxa Smartio/Industio Configuration Panel».

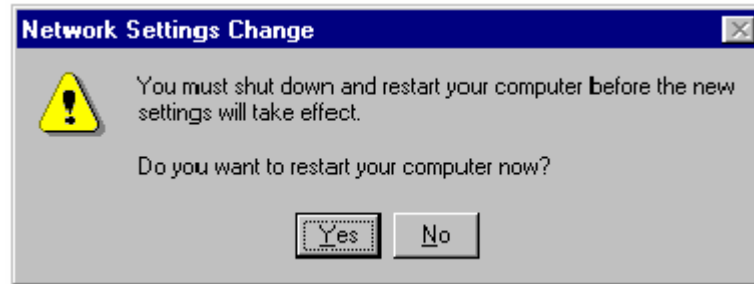
**Обратите внимание:** если вам нужно установить более одной платы, нажмите на кнопку «**Add**» и повторите шаги **6 – 8**, чтобы настроить другие платы. В одной системе может быть установлено до четырех плат серии Smartio C168.

Нажмите на кнопку **[OK]**, чтобы завершить настройку.



9. Когда конфигурация завершена, нажмите кнопку **[ОК]**, чтобы выйти из диалога «Network Settings».

10. Перезагрузите систему Windows NT.



**Обратите внимание!** Новая конфигурация драйвера не вступит в силу до перезапуска системы.

11. После перезагрузки системы вы можете просмотреть лог файл, чтобы убедиться в полной инициализации портов платы.

- Запустите программу **Event Viewer** [Просмотр События] и войдите в пункты **[Log]** и **[System]**, чтобы проверить наличие сообщения об успешной установке наподобие **“MOXA C168 series, with first serial port COM3, has been enabled”** для каждой настроенной платы.
- Если появляется сообщение об ошибке наподобие **“Cannot find any configured MOXA Smartio C168 series board!”**, то обратитесь к главе «Решение проблем».

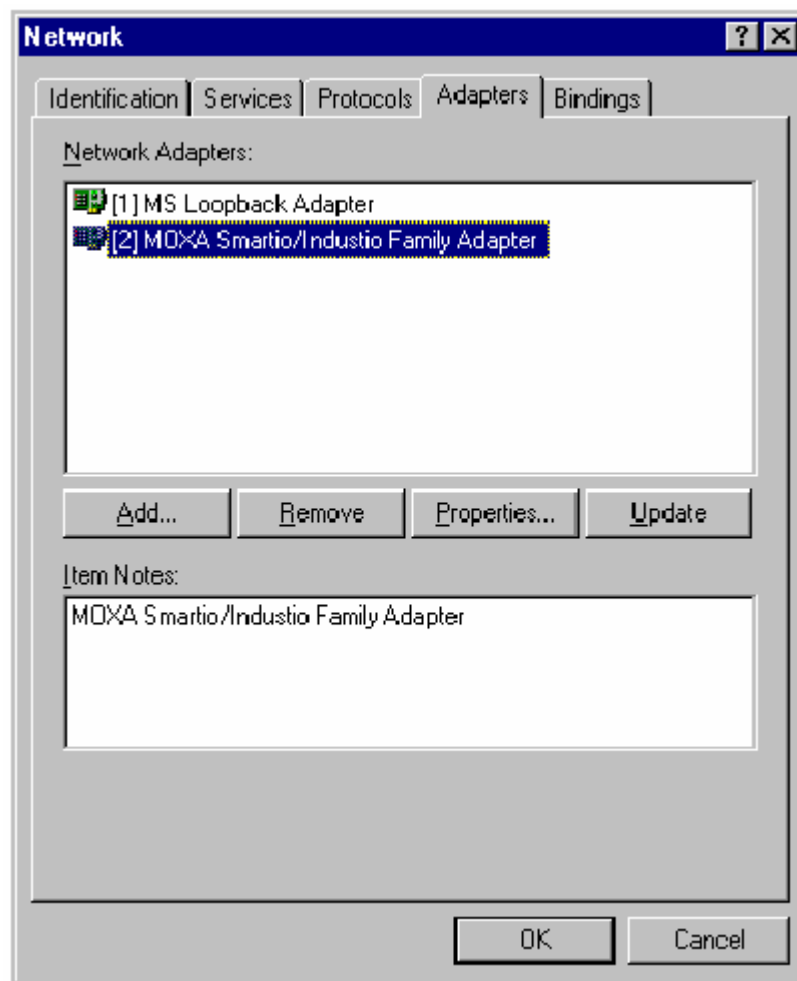
**Обратите внимание!**

Как только плата и драйвер корректно установлены, вы можете начинать разработку приложений с помощью библиотеки PCOMM (см. главу «Программное обеспечение») или Microsoft Win32 API. Также имеются готовые приложения, такие, как утилита Terminal emulator (см. главу «Программное обеспечение») или HyperTerminal для обмена данными и работы с Сервисом Удаленного Доступа.

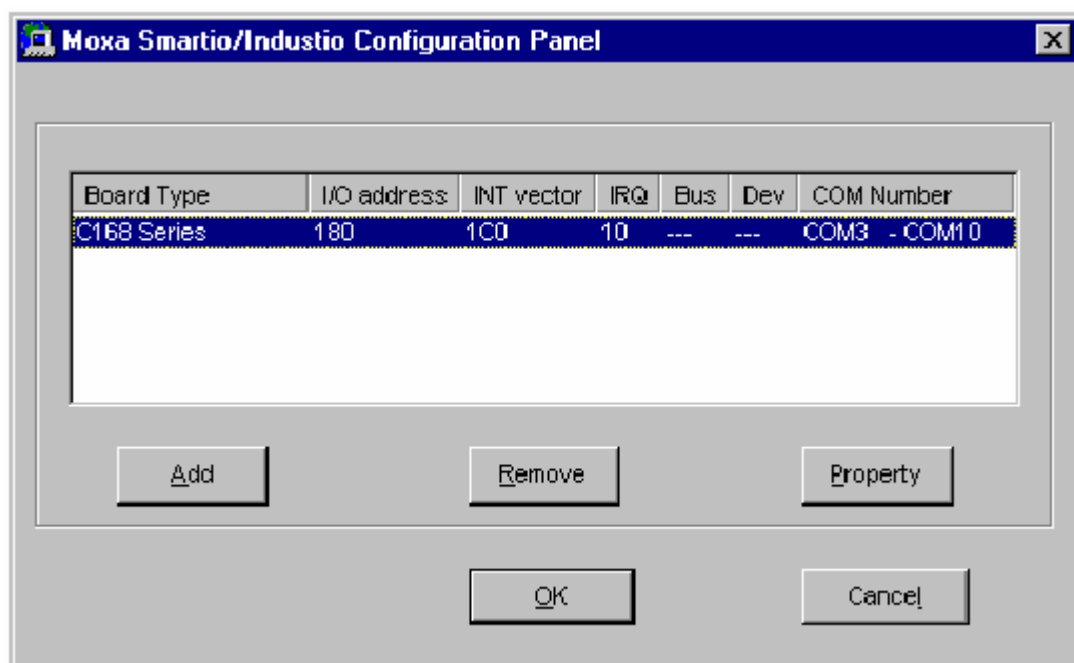
## Настройка платы и портов

Если драйвер уже установлен и необходимо изменить настройки портов, тогда проделайте следующую процедуру.

1. Откройте [**Control Panel**], нажмите иконку [**Network**] и выберите закладку [**Adapters**].
2. Нажмите на пункт “**MOXA Smartio/Industio Family Adapter**” в списке «Network Adapters».



3. Нажмите на кнопку **[Property]**, чтобы открыть диалоговое окно «**Moxa Smartio/Industio Configuration Panel**». Далее см. шаги 6 – 10 предыдущего раздела «Установка драйвера».



В этой конфигурационной панели вы можете:

- Нажать на кнопку [**Property**], чтобы открыть диалог **Property**. В нем вы можете корректировать значения полей “COM Number”, “INT Vector”, “Interrupt no” и “Base I/O Port Address”.  
Более подробно этот процесс описывают **шаги 6 – 8** предыдущего раздела, за исключением того, что поле “Board Type” не изменяется.
- Нажать [**Add**], чтобы добавить еще не настроенные в системе платы. См. **шаги 6 – 8** предыдущего раздела «Установка драйвера».
- Нажать кнопку [**Remove**], чтобы удалить выбранную в списке плату.
- Нажать кнопку [**OK**], подтверждая сделанные изменения.
- Нажать кнопку [**Cancel**], чтобы выйти из диалога с неизменной конфигурацией.

## Обновление драйвера

Чтобы обновить драйвер платы серии Smartio C168H, просто удалите драйвер, как описано в следующем разделе, а затем переустановите его как сказано в разделе «**Установка драйвера**».

## Удаление драйвера

Чтобы удалить драйвер платы серии Smartio C168:

1. Откройте [**Control Panel**], нажмите иконку [**Network**] и выберите закладку [**Adapters**].

2. Выберите пункт «**MOXA Smartio/Industio Family Adapter**» в списке устройств, а затем нажмите кнопки [**Remove**] и [**OK**], чтобы удалить драйвер.
3. Перезагрузите систему, чтобы активизировать новую конфигурацию.

## Windows 95/98

Windows 95/98 поддерживает до **128** последовательных портов, от **COM1** до **COM128**. Для максимальной реализации возможностей системы Windows 95/98, многозадачности и работы с параллельными потоками, для плат серии Smartio C168 и других плат MOXA разработаны специальные 32-разрядные драйвера (VxD) с поддержкой коммуникационного драйвера (VCOMM). Драйверы соответствуют стандарту Win32 API COMM.

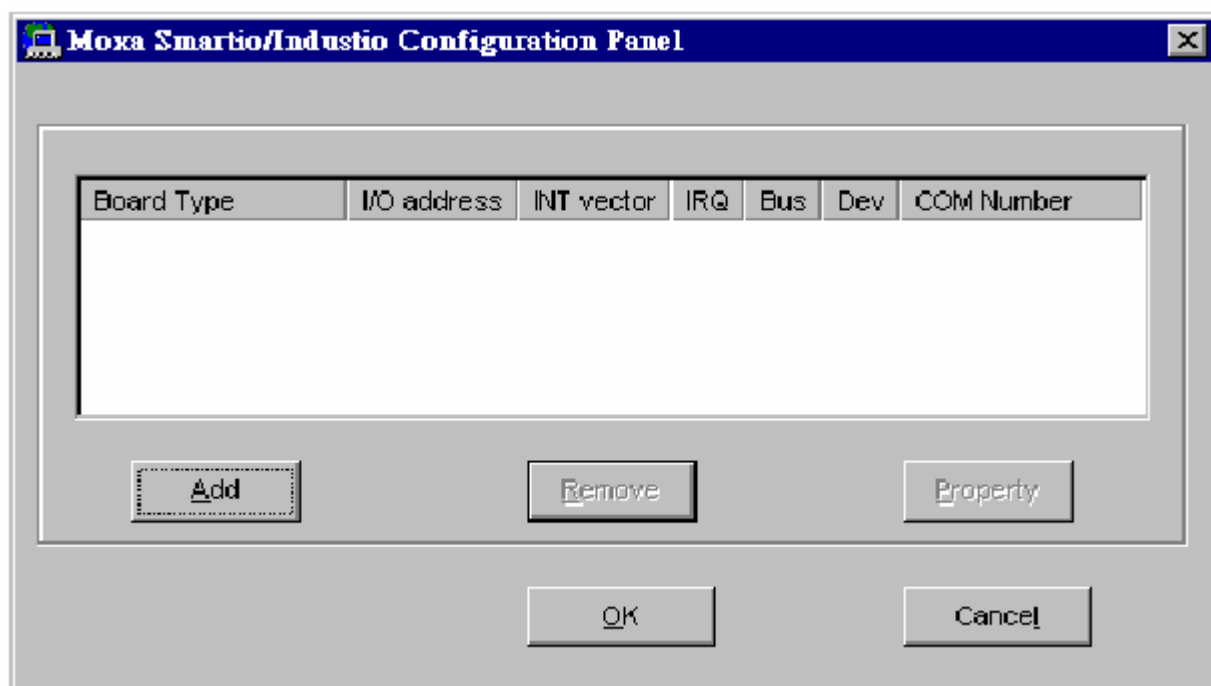
- Если Вы впервые устанавливаете драйвер, обратитесь непосредственно к разделу "Установка драйвера".
- Если хотите перенастроить драйвер и порты установленной платы добавить новые или удалить имеющиеся платы, обратитесь к разделу "Настройка платы и портов".
- Чтобы обновить или удалить драйвер, пожалуйста, прочтите раздел "Обновление драйвера" или "Удаление драйвера".

### Установка драйвера.

Если позволяют ресурсы системы, то вместе могут быть установлены до четырех плат серии Smartio C168.

Далее описывается процедура **первичной** инсталляции платы серии Smartio C168 в систему под Windows 95/98.

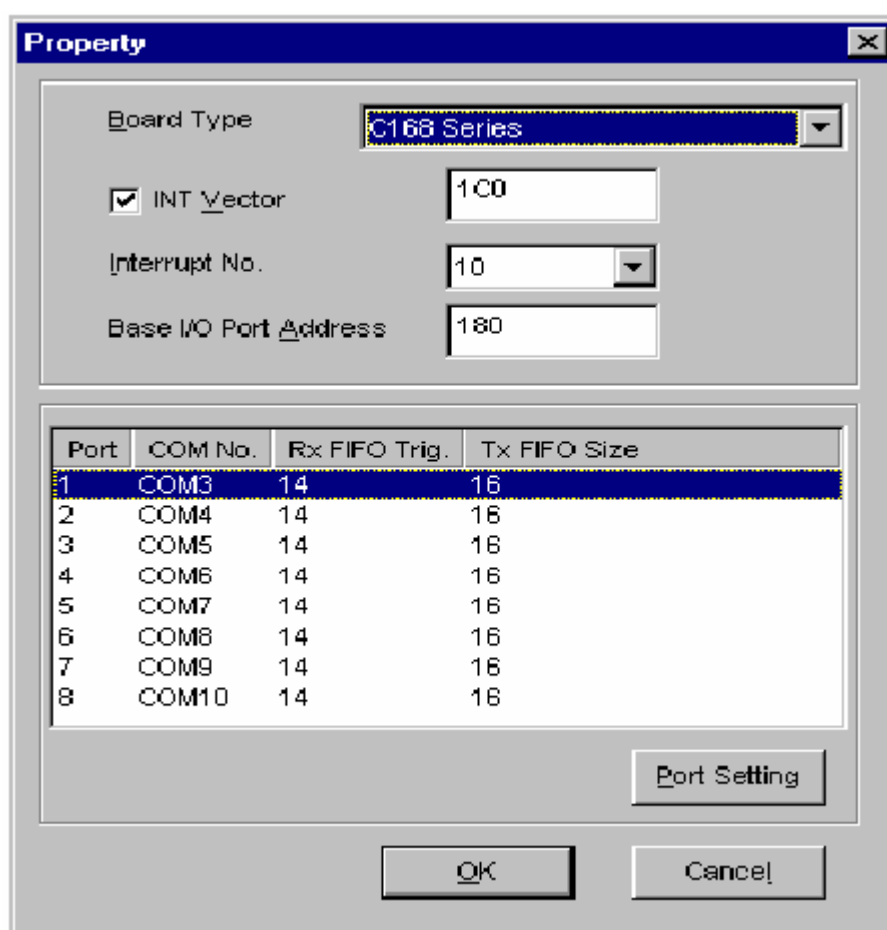
1. Запустите файл **Setup95.exe** с диска драйверов MOXA.
2. Щелкните кнопку **[Next>]** в диалоговом окне "Welcome ... ". Затем нажмите кнопку **[Next>]** в диалоговом сообщении "Ready ...".
3. Нажмите кнопку **[Finish]** в диалоговом сообщении "Complete ... ", чтобы вывести конфигурационную панель.
4. В появившемся диалоговом окне «**MOXA Smartio/Industio Family Configuration Panel**» вы можете настроить установки платы и портов.



5. В диалоге «**Moxa Smartio/Industio Configuration Panel**» нажмите на кнопку **Add**, открывая диалог **Property**, чтобы добавить новую плату серии Smartio C168. В поле «Board Type» выберите пункт «**C168 Series**». При необходимости, установите желаемое значение вектора прерывания в поле «INT Vector». Выберите нужный номер прерывания в поле «Interrupt No» и номер адреса ввода-вывода в поле «Base I/O Port Address». Все эти настройки

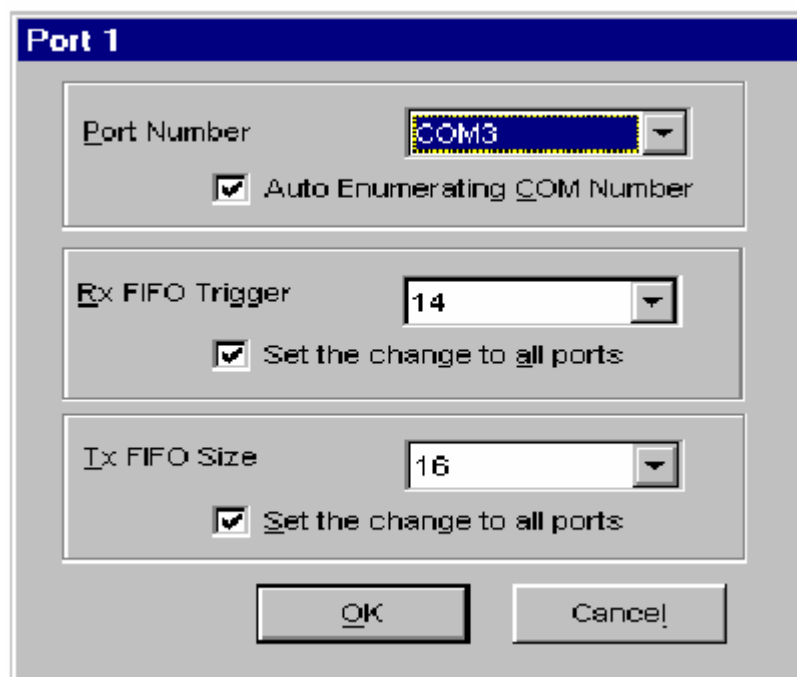


должны совпадать с аппаратными настройками платы и не конфликтовать с другими устройствами.



**Обратите внимание:** если у вас нет необходимости изменять другие какие-либо настройки, то переходите к **шагу 7**.

6. В диалоге «**Property**» выберите из списка желаемый порт и нажмите кнопку [**Port Setting**], чтобы открыть диалоговое окно «**Port #**», позволяющем изменить COM номера или настройки FIFO.



- **Номер порта**

Вы должны установить все порты платы с желаемыми номерами **COM**, которые не должны противоречить другим уже занятым номерам COM. В этом окне есть два пути придания физическим портам соответственных COM номеров в зависимости от переключателя "**Auto Enumerating COM number**".

Если установлен флажок "Auto Enumerating COM number", то определите номер первого порта, то остальным будут присвоены последовательно возрастающие номера COM. Например, если первый порт отображен как COM3, то второй порт отображается соответственно как COM4.

Если флажок "Auto Enumerating COM number" не установлен, то вы должны вручную определить номер для каждого COM порта. Например, второй порт может быть обозначен как COM10, в то время как первый порт отображается как COM3.

- **Rx FIFO Trigger**

Rx FIFO может принимать следующие значения: 1, 4, 8 или 14 байт; по умолчанию задается значение 14 байт.

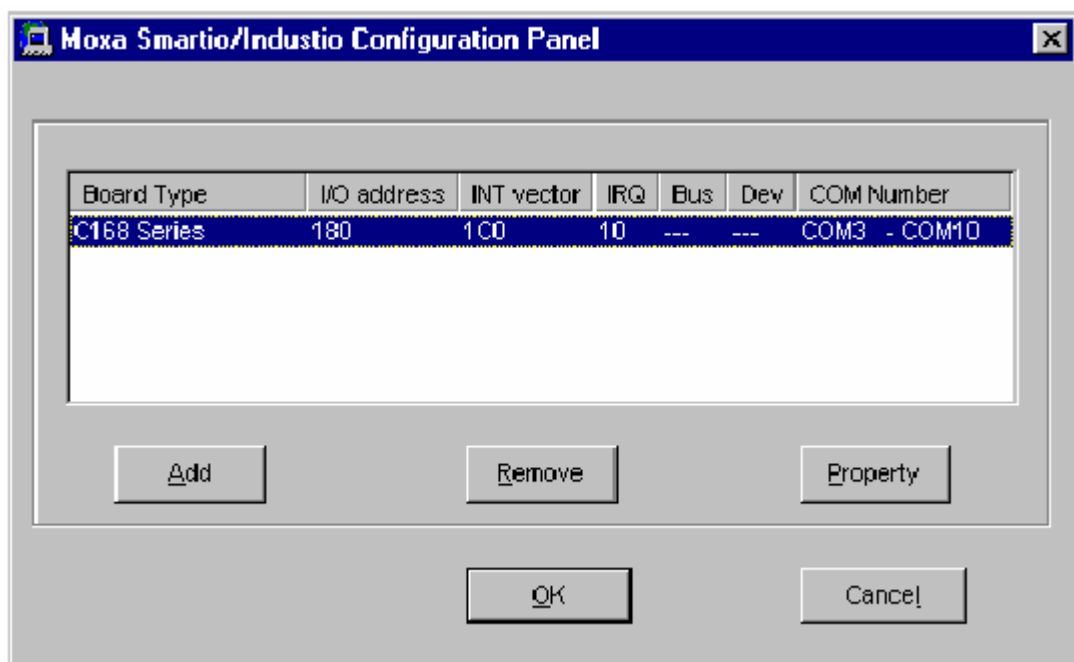
- **Tx FIFO Size**

Допустимое значение Tx FIFO – от 1 до 16 байт. По умолчанию задается 16 байт.

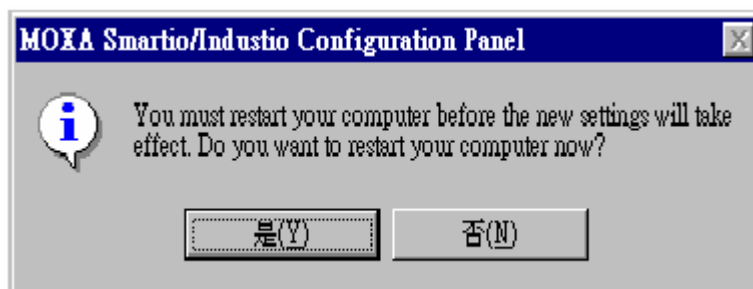
7. Нажмите кнопку **[OK]** в диалогах «Port #» и «Property», чтобы вернуться к окну «Moxa Smartio/Industio Configuration Panel».

**Обратите внимание:** если вам нужно установить более одной платы, нажмите на кнопку «**Add**» и повторите шаги **5 – 7**, чтобы настроить другие платы. В одной системе может быть установлено до четырех плат серии Smartio C168.

Нажмите на кнопку **[OK]**, чтобы завершить настройку.



8. Перезагрузите систему Windows 95/98.



**Обратите внимание!** Новая конфигурация драйвера не вступит в силу до перезапуска системы.

9. После перезагрузки системы могут появиться все возможные сообщения об ошибках. Если их нет, значит установка прошла успешно

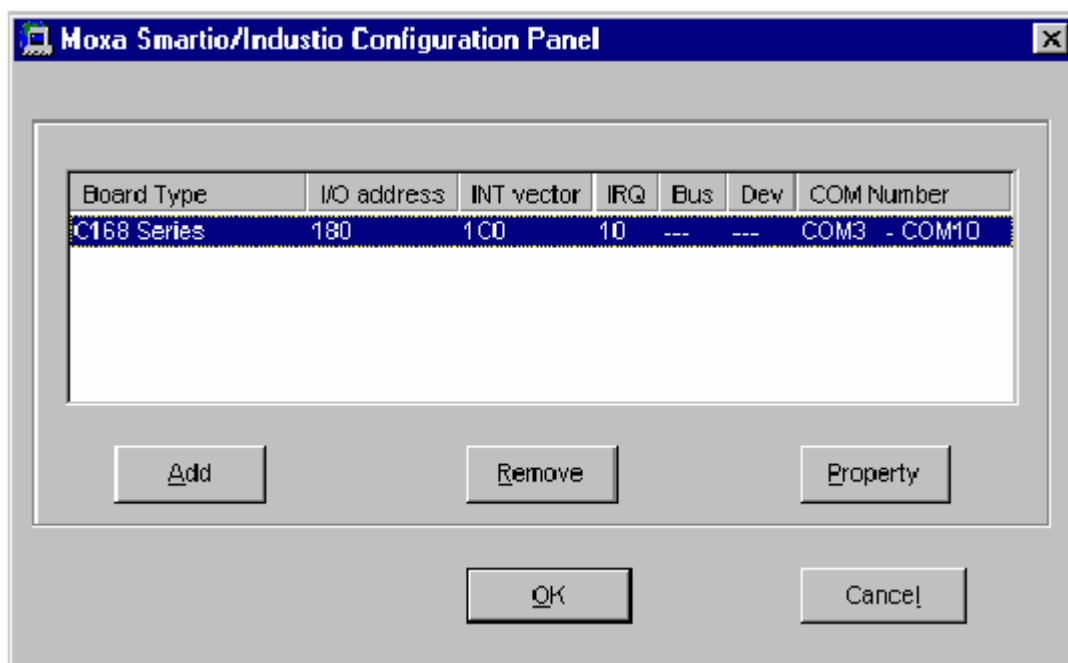
Если появляются сообщения типа **“Smartio C168 Series (CAP=0x0180, port 1=COM3): Board is not found”**, то обратитесь к главе «Решение проблем».

**Обратите внимание!** Как только плата и драйвер корректно установлены, вы можете начинать разработку приложений с помощью библиотеки PCOMM (см. главу «Программное обеспечение») или Microsoft Win32 API. Также имеются готовые приложения, такие, как утилита Terminal emulator (см. главу «Программное обеспечение») или HyperTerminal для обмена данными и работы с Сервисом Удаленного Доступа.

## Настройка платы и портов

Если драйвер уже установлен и необходимо изменить настройки портов платы серии Smartio C168, добавить или удалить существующие в системе платы под Windows 95/98, тогда проделайте следующую процедуру.

1. Нажмите кнопку [**Start**], затем выберите пункт [**Programs**], [**MOXA Utilities**] и выберите значок [**Moxa Smartio/Industio Configuration Panel**].
2. Появится панель «Moxa Smartio/Industio Configuration Panel». Далее см. шаги 5-7 предыдущего раздела «Установка драйвера».



В этой конфигурационной панели вы можете:

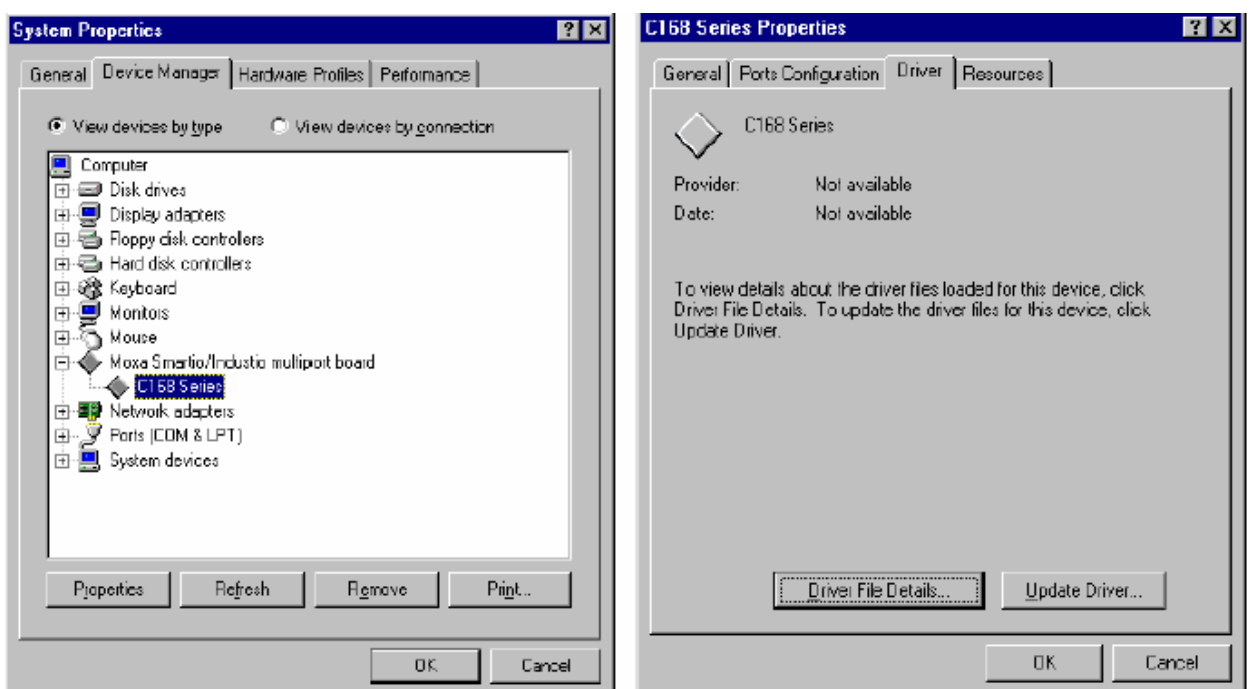
- Нажать на кнопку [**Property**], чтобы открыть диалог **Property**. В нем вы можете корректировать значения полей “COM Number”, “INT Vector”, “Interrupt no” и “Base I/O Port Address”.

Более подробно этот процесс описывают **шаги 5 – 7** предыдущего раздела, за исключением того, что поле “Board Type” не изменяется.

- Нажать [**Add**], чтобы добавить еще ненастроенные в системе платы. См. **шаги 5 - 7** предыдущего раздела «Установка драйвера».
- Нажать кнопку [**Remove**], чтобы удалить выбранную в списке плату.
- Нажать кнопку [**OK**], подтверждая сделанные изменения.
- Нажать кнопку [**Cancel**], чтобы выйти из диалога с неизменной конфигурацией.

## Обновление драйвера

Откройте иконку «**Control Panel**» [Панель управления], затем иконку **System** [Система] и выберите вкладку **Device Manager** [Устройства]. Затем выберите и откройте опцию “**Moxa Smartio/Industio Multiport Board**”, а затем “**C168 Series**”. Нажмите на кнопку **Properties** [Свойства], а затем выберите вкладку **Driver** [Драйвер] и нажмите на кнопку **Update Driver** [Обновить драйвер].



## Удаление драйвера

Откройте [Control Panel], дважды нажмите на иконку [Add/Remove Programs] и выберите закладку [Install/Uninstall]. Выберите опцию «MOXA Smartio/Industio Driver», а затем нажмите кнопку [OK], чтобы удалить драйвера.



# DOS

MOXA DOS API -232 - это пакет программ, который помогает пользователям разрабатывать и/или отлаживать программы для осуществления последовательной связи. Далее показано, как работать с этим пакетом, чтобы установить, загрузить или выгрузить драйвер.

Более подробно программирование с помощью библиотеки API-232 и утилиты описаны в следующей главе "Программное обеспечение".

## Установка драйвера

Запустите с диска инсталляционную программу **DOSINST.EXE**. Определите каталог для API-232 (например. **C:\MOXA**), куда будет скопирован программный драйвер. Нажмите **F2**, чтобы запустить инсталляцию.



После того, как инсталляция закончена, запустите программу установки.

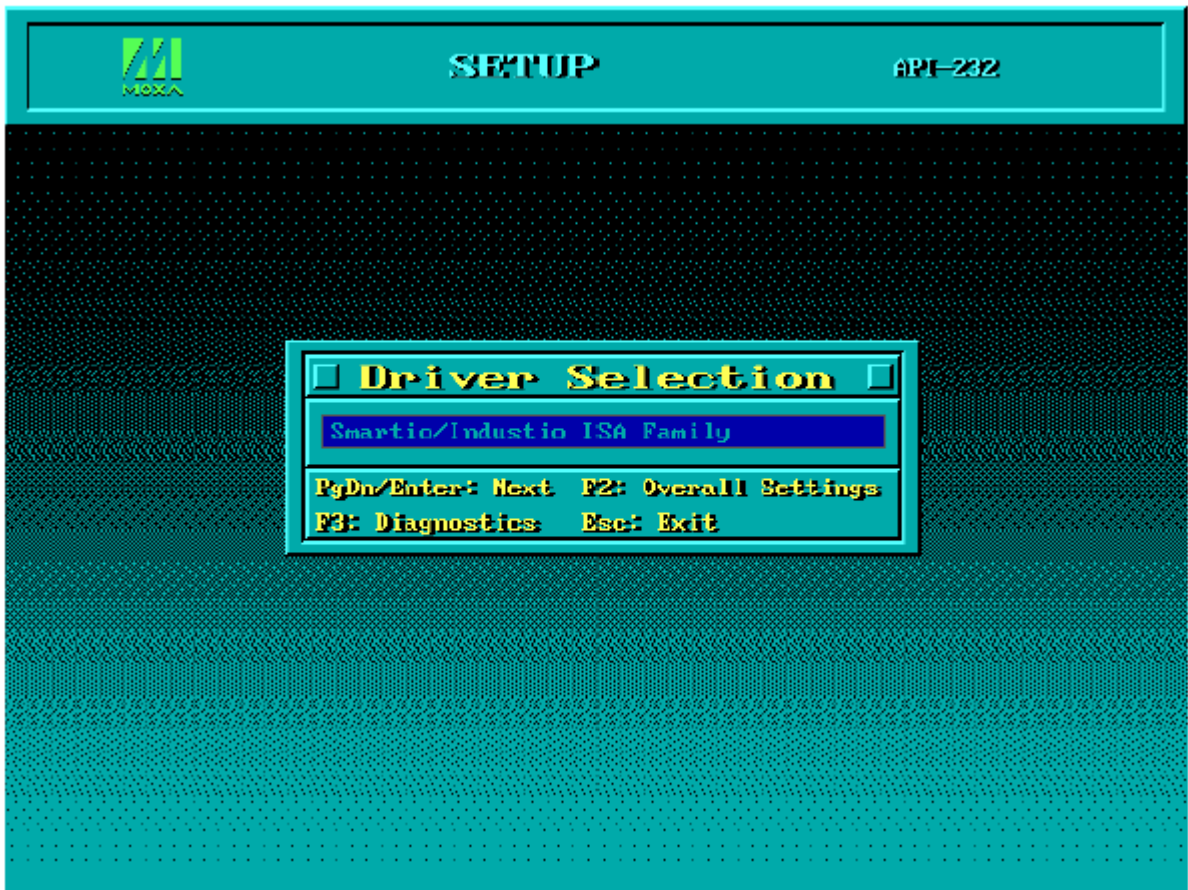




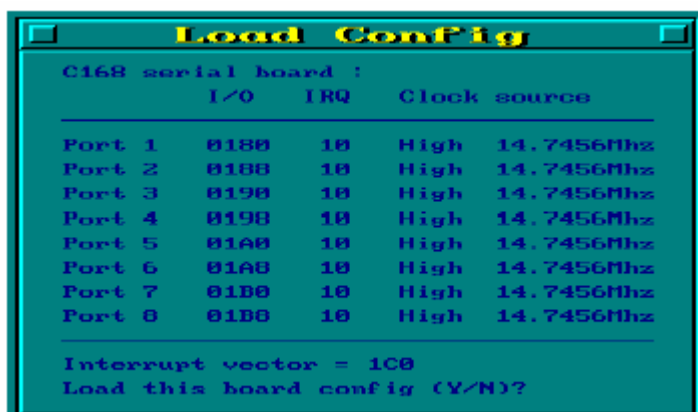
## Конфигурирование драйвера.

Далее следует пошаговое описание процесса конфигурирования драйвера платы серии Smartio C168 и. Обратите внимание, что здесь не приводится полное описание программы конфигурации. Пожалуйста, нажмите кнопку F1 для вызова интерактивной инструкции по установке программы.

1. Запустите программу установки, `\BIN\SETUP.EXE`, из каталога API-232. Выберите пункт "**Smartio/Industio ISA Family**" в диалоговом окне "Driver Selection".



2. Нажмите **Enter** в появившемся диалоговом окне SETUP, затем кнопку **F8**, чтобы определить **CAP адрес**; затем нажмите **ENTER** и «Y»(**YES**), загружая конфигурацию настраиваемой платы.



3. После этого в соответствующих полях будут отражены другие заданные по умолчанию параметры, такие как номер порта, размер буфера и т. д.

**Обратите внимание!** На этом этапе вы можете завершить установку платы серии Smartio C168. Если вы не хотите изменять какие-либо другие настройки платы, то можете перейти к **шагу 5**.



**Smartio/Industio ISA Family**

Port number	01	02	03	04	05	06	07	08
Base I/O address	180	188	190	198	1A0	1A8	1B0	1B8
Interrupt number	10	10	10	10	10	10	10	10
TxD buffer size	1K	1K	1K	1K	1K	1K	1K	1K
RxD buffer size	1K	1K	1K	1K	1K	1K	1K	1K
Baud rate	9600	9600	9600	9600	9600	9600	9600	9600
Character length	8	8	8	8	8	8	8	8
Stop bits	1	1	1	1	1	1	1	1
Parity	None	None	None	None	None	None	None	None
DTR output state	On	On	On	On	On	On	On	On
RTS output state	On	On	On	On	On	On	On	On
CTS flow control	No	No	No	No	No	No	No	No
RTS flow control	No	No	No	No	No	No	No	No
Tx MON/OFF ctrl	No	No	No	No	No	No	No	No
Rx MON/OFF ctrl	No	No	No	No	No	No	No	No

**F1: Help      F2: Overall Settings    F3: Add port      F4: Delete port**  
**F5: Group edit    F6: INT vector    F8: Load config    F10: Save    Esc: Abort**

Теперь вы можете вводить/изменять настройки каждого порта. Отображаемые здесь значения являются изначальными настройками.

### Условные обозначения

Ниже приводится описание некоторых важных полей и функций.

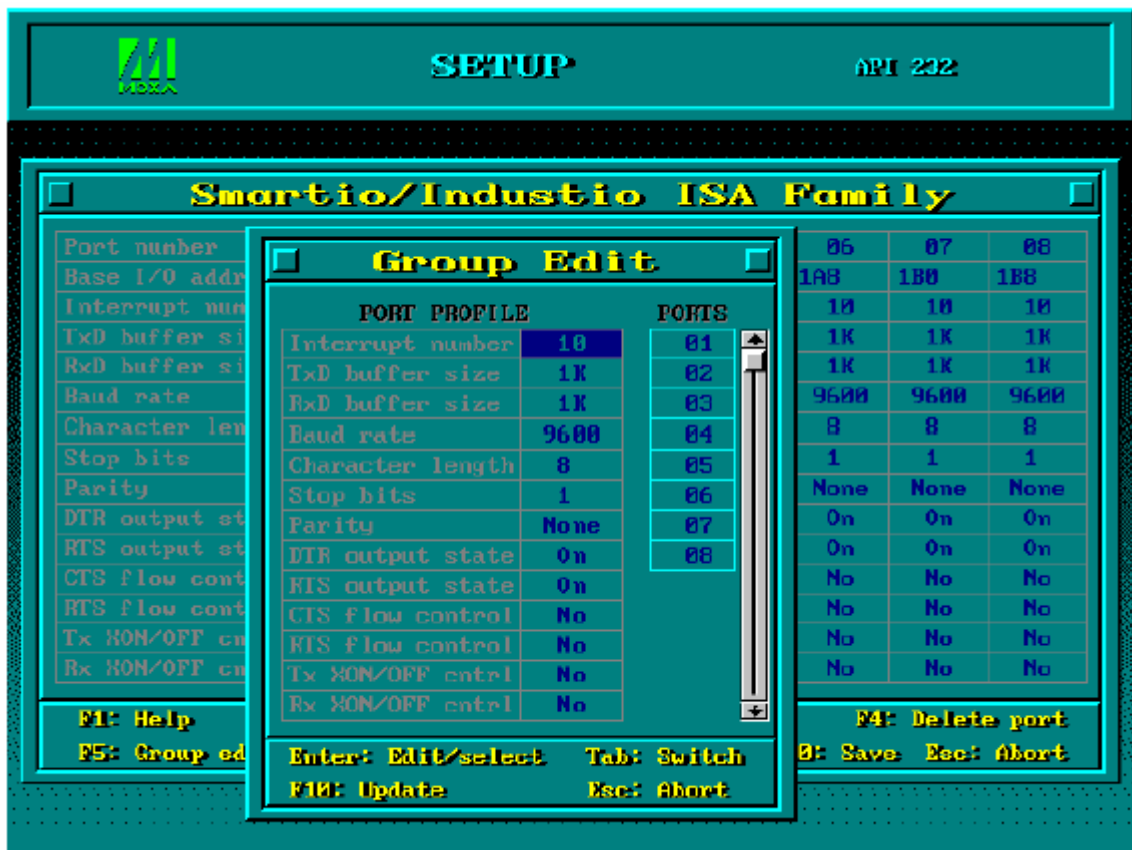
**Port Number:** Это идентификатор каждого порта. Программы обращаются к порту по присвоенному ему номеру (ID). Назначение одинаковых номеров разным портам не допустимо. То есть, каждый последовательный порт MOXA в последовательном программировании определяется как номер порта.

Порту может быть присвоено любое значение в интервале от 0 до 255, исключая повторения и неопределенные значения. В общем, при определении номеров портов платы вы можете исходить из удобства в дальнейшей работе с программами.

**TxD buf size:** Буфер передачи (вывод), определяемый в системе для каждого порта.

**RxD buf size:** Буфер приема (ввод), определяемый в системе для каждого порта.

**F5: Group Edit:** Эта удобная функция поможет Вам отредактировать конфигурацию нескольких портов одновременно.



**F6: INT Vector:** Здесь вы можете установить значение вектора прерывания для каждого порта. Если вы установите значение «Yes» (по умолчанию) для этой функции, то будет обеспечиваться наилучшая производительность платы.



4. Нажмите кнопку **F10**, чтобы сохранить последнюю конфигурацию и выйти из программы SETUP.

## Загрузка драйвера

Завершив установку, Вы можете загрузить драйвер, написав в командной строке DOS "**BIN\MX-DRV.EXE**". Драйвер автоматически обнаружит плату Smartio C168. Если плата(ы) обнаружена(ы), появляется сообщение:

```
API -232 Version 3.5
Universal 2/4/8 serial ports Communication Driver
Setup driver ...
Device driver setup O.K.
```

Это означает, что драйвер платы серии Smartio C168 установлен правильно. С этого момента Вы готовы к работе с приложениями,

поддерживающими функции API-232, или началу разработки нового приложения с помощью специальной библиотеки API-232.

Если что-то пошло не так, например, плата не соответствует конфигурации или отсутствует, на экране появляется сообщение:

```
API-232 Version 3.5
Universal 2/4/8 serial ports Communication Driver
Setup driver ...
None serial port found!!
```

Это означает, что драйвер платы серии Smartio C168 установлен неправильно. Тогда обратитесь за разъяснением причин к главе “Решение проблем”.

## Выгрузка драйвера

Чтобы выгрузить драйвер Smartio C168 из памяти, напечатайте в командной строке DOS “**SER-DRV /Q**”.

## UNIX

Существуют различные операционные системы UNIX, такие как SCO UNIX, UNIX SVR4.2, Solaris и т.д. Для различных систем UNIX требуются различные типы драйверов. В настоящее время MOXA выпускает драйверы устройства, поддерживающие системы **SCO UNIX/Open Server** и **UNIX SVR4.2**.

В этой главе описана процедура инсталляции драйвера. Также будет рассмотрена административная утилита **moxaadm**, которая используется для настройки, контроля и эмуляции терминала; с ее же помощью регулируются некоторые связанные с установкой аспекты типа device naming, baud rate setting и terminal.

Если Вы заинтересованы в программировании под UNIX с использованием `ioctl()` команд, ознакомьтесь с главой "Программное обеспечение". Решение проблем, специфичных для UNIX, включено в главу "Решение проблем".

## Установка драйвера.

Далее описывается процесс установки драйвера платы серии Smartio C168 под SCO UNIX/OpenServer и UNIX SVR4.2.

**Шаг 1.** Войдите в систему как `root`.

**Шаг 2.** Перейдите в корневой каталог, введя следующую команду:

```
# cd /
```

**Шаг 3.** Вставьте дискету драйвера в 1.44MB дисковод для гибких дискет, A: или B:. Если файлы драйвера получены иным путем, поместите их в каталог `/tmp/moxa` и **переходите к шагу 5.**

**Шаг 4.** Введите следующую команду, чтобы разархивировать файл установки

```
# tar xvf /dev/fd0135ds18 /tmp/moxa/mxinstall (если  
1.44MB A:)
```

**Шаг 5.** Запустите инсталляцию программы

```
# /tmp/moxa/mxinstall
```

Следуйте инструкциям, чтобы закончить инсталляцию драйвера:



Из предложенного списка вам нужно будет выбрать необходимую операционную систему. MOXA поддерживает SCO UNIX (SCO OpenServer и SCO Open Desktop) и UNIX SVR4.2 (UnixWare). У вашего дилера вы можете запросить новые драйвера поддержки. Если вашей системы нет в перечне, то попробуйте выбрать аналогичную.

*Copyright (C) 199x MOXA Technologies Co., Ltd. All Rights Reserved.*

*MOXA UNIX Device Driver Installation Ver. x.x*

Пожалуйста, выберите одну из следующих операционных систем:

1. SCO UNIX
2. UNIX SVR4.2

Select:

Пожалуйста, выберите устройство, в котором располагается диск/дискета с драйверами:

1. /dev/fd0135ds18 (A: 1.44MB)
2. /dev/fd096ds15 (A: 1.2 MB)
3. /dev/fd1135ds18 (B: 1.44MB)
4. /dev/fd196ds15 (B: 1.2 MB)
5. Hard Disk /tmp/moxa

Select:

Затем появляется диалоговое окно инсталляционной **утилиты мультипортовой платы MOXA** для установки драйвера.

Выберите **драйвер серии C168** и следуйте дальнейшим инструкциям.

**Шаг 6.** Настройте плату с помощью

## # moxaadm

Утилита moxaadm используется для определения количества плат Smartio C168, установленных в системе UNIX, распознавания их основных настроек (адреса ввода-вывода, IRQ) и т.д. Эту утилиту необходимо запускать каждый раз, когда эти настройки изменяются, чтобы сообщить об этом драйверу UNIX. Для получения более подробной информации ознакомьтесь с разделом «Административная утилита moxaadm».

Эта утилита не может изменять аппаратные настройки платы Smartio C168, такие как адреса ввода-вывода или IRQ. Если вам нужно изменить аппаратную конфигурацию, то запустите в среде DOS утилиту IO-IRQ, которая содержится на диске с драйверами под Dos/Windows3.x.

При использовании драйверов MOXA в одной системе UNIX может быть установлено до двух плат серии Smartio C168.

**Шаг 7.** Теперь вы можете перезапустить систему. Если у вас возникнут проблемы с загрузкой нового ядра, используйте сохранение последней его рабочей копии (**unix.moxa** или **/stand/unix.moxa**), чтобы восстановить работоспособность системы в исходном варианте. За более подробной информацией обратитесь к главе «Решение проблем».

После повторного запуска системы, появятся сообщения, описанные ниже:

**SCO UNIX, OpenServer**

“C168 0x0180-0x01BF 11 Ver = x.x type=C168H (high speed)”

### **UNIX SVR4.2**

“C168H board, base address 0x180, irq 5, Ver. x.x (high speed)”,

если порт настроен в режиме высокой скорости. В обратном случае порт находится в режиме нормальной скорости.

Если появляются сообщения об ошибке:

### **SCO UNIX/OpenServer,**

“WARNING! C168 board IRQ at 11 mismatch (base address at 0x180)!”

### **UNIX SVR4.2,**

“WARNING! C168 board IRQ at 11 mismatch (base address at 0x180)!”

то это свидетельствует о том, что различается значение IRQ в аппаратных и программных настройках. Этот порт не будет работать, но остальные порты продолжат функционировать.

Если появляются следующие сообщения об ошибке:

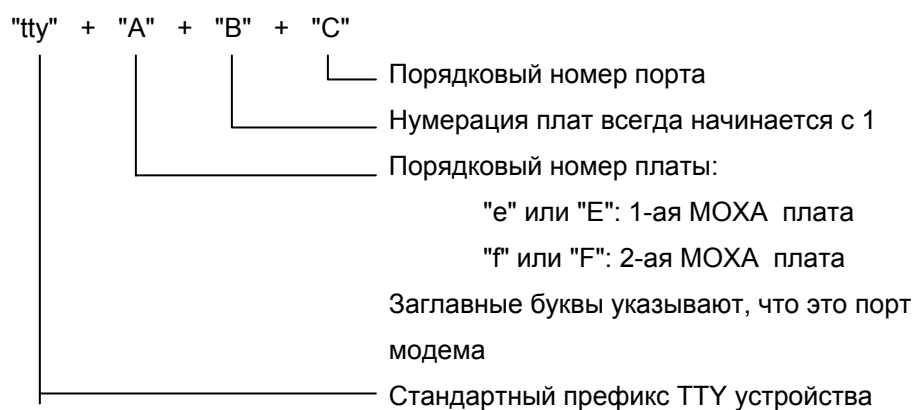
“WARNING! C168 board base address at 0x180 not found!” или “WARNING! C168 board interrupt vector disabled (Board base = 0x180 )!”, то это означает, то не обнаружен адрес фиксированной памяти или неправильно задан вектор прерывания. В этом случае не будут работать все порты.

Подробности смотрите в главе «Решение проблем».

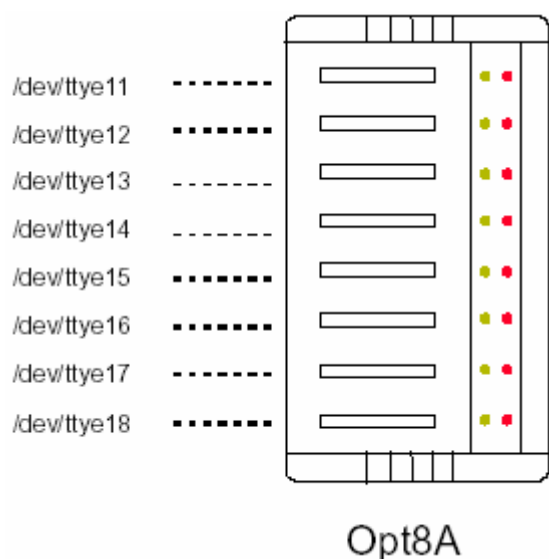
## Назначение имен устройств MOXA TTY

Если Smartio C168 полностью правильно установлена, то для каждого порта в каталоге `/dev` будут созданы два устройства TTY: одно для соединения с **модемом** (например, `ttyE11`), другое для **прочих устройств** (например, `ttye11`). Эти два устройства назначаются на один и тот же порт, за исключением того, что модемный TTY проверяет наличие DCD сигнала, что необходимо для автоматического прекращения работы, когда DCD сигнала нет.

Назначение устройств MOXA TTY: `/dev/tty {e-f} {1} {1-8}`, где:



Например:



## Настройки скорости

Для плат серии Smartio C168, настроенных в **высокоскоростном режиме (High Speed Spectrum)**, скорость реальной работы будет в восемь раз превышать скорость, отображаемую при помощи команды «**stty**». Например, если порт, работающий в нормальном режиме (Normal Speed Spectrum) показывает 38.4 Kbps, то его реальная скорость также будет 38.4 Kbps; если же задана работа в высокоскоростном режиме (High Speed Spectrum), то реальная скорость будет 307.2 Kbps (38.4 Kbps x 8).

Обратите внимание, что обозначение **B50** присваивается скорости 57600 bps, а **B75** – 115.2 Kbps. Кроме этого, если плата серии Smartio C168 настроена на работу в **высокоскоростном** режиме, то реальная рабочая скорость будет превышать отображаемую в восемь раз. Следовательно обозначению B50, 57600 bps, будет соответствовать реальная скорость 460.8 Kbps, а B75, 115.2 Kbps, - 921.6 Kbps.

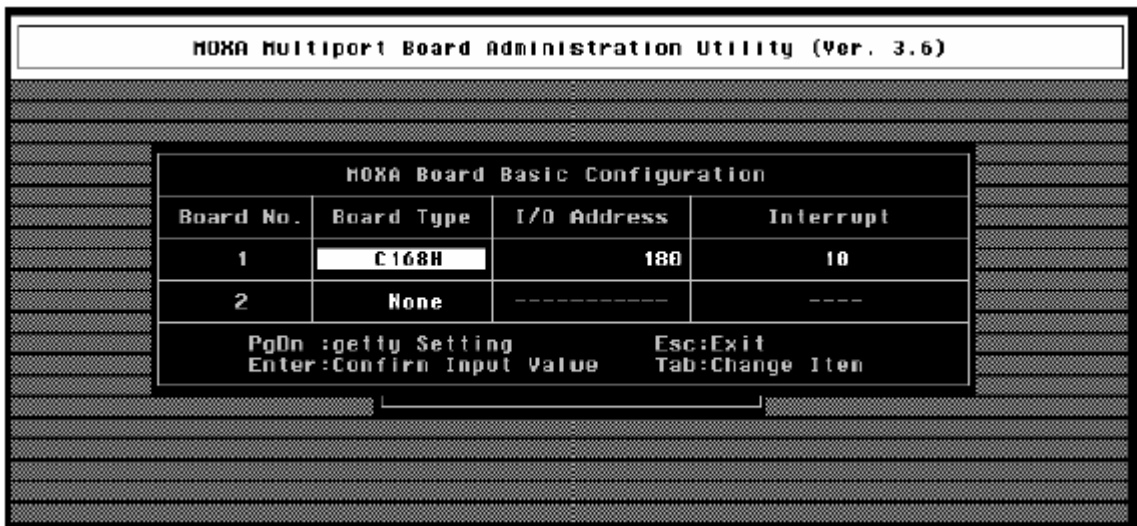
## Административная утилита "mohaadm"

Вы можете использовать утилиту **mohaadm** для изменения основных и дополнительных настроек Smartio C168, для контроля за активными портами, работы с эмулятором терминала и удаления установленного драйвера MOXA из системы.

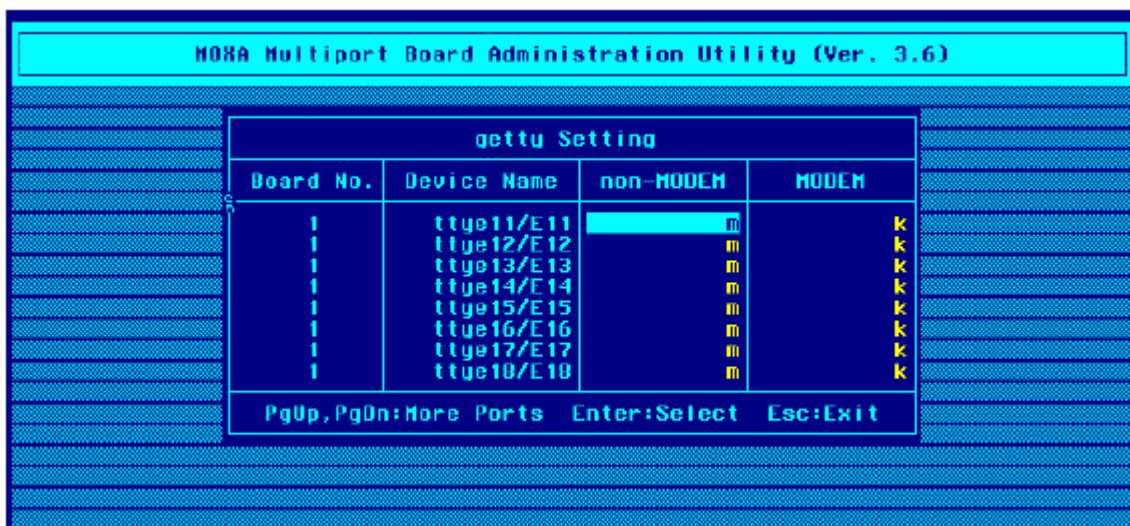


## Basic configuration

В поддиалоге [Basic configuration] вы можете установить значения адреса ввода-вывода и прерывания, которые должны соответствовать аппаратным настройкам. **Ядро необходимо перезагружать** после любого изменения этих настроек.



В поддиалоге [getty Setting] есть одно важное поле.



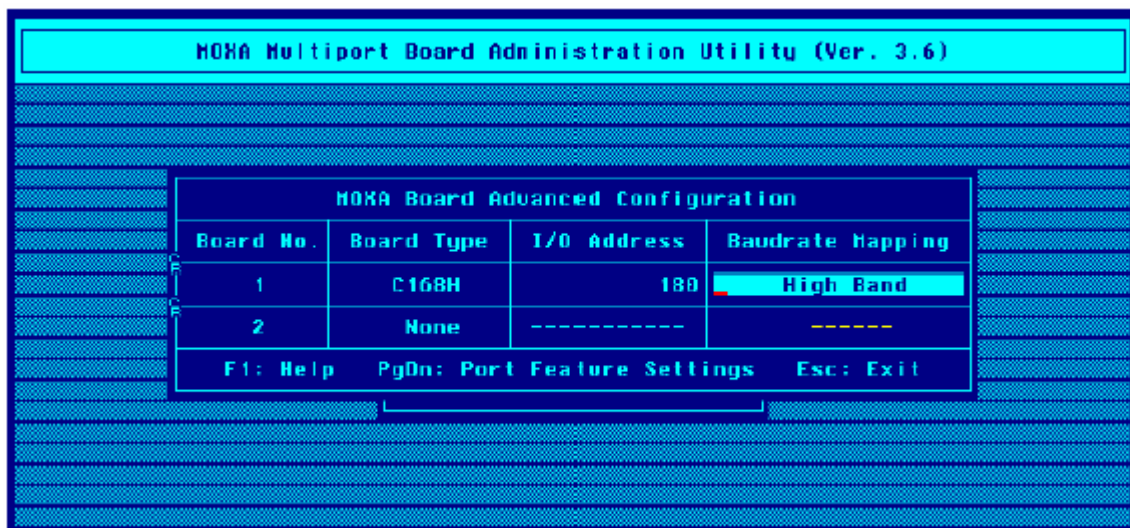
## Установка скорости для модемных и немодемных устройств

Это поле устанавливает начальную скорость и параметры для модемных и немодемных соединений. При этом просто устанавливаются параметры getty в системном файле /etc/inittab, который может быть так же изменен вручную системным администратором. Эти параметры переписываются из системного файла UNIX "getty default", то есть, "/etc/gettydefs". При необходимости измените это поле.

В некоторых случаях может возникнуть необходимость изменить файл «getty default». Например, в некоторых UNIX, символ "9600" обозначает 8-битные данные без проверки четности в то время как в других это же обозначение может показывать 7-битные данные с проверкой четности. Поэтому, пожалуйста, тщательно проверьте соответствие установок в обоих файлах, иначе будут неизбежны ошибки в данных.

## Advanced Configuration

В диалоге [Advanced configuration] пользователь может установить значения baud rate mapping, Rx FIFO trigger level, Tx FIFO trigger level и RTS/CTS hardware flow control.



### Baud rate Mapping

Это поле может принимать два значения: High Band и Low Band. Максимальное значение скорости при установке значения High Band - 921.6 Kbps, а для Low Band – 38.4 Kbps. Из-за ограничения UNIX termio/termios definition, обозначение максимальной скорости в бодах – B38400(соответствует 38.4 Kbps). Если пользователь хочет использовать большую скорость при работе с платой MOXA, то значения baud rate будут изменены согласно следующей таблице:

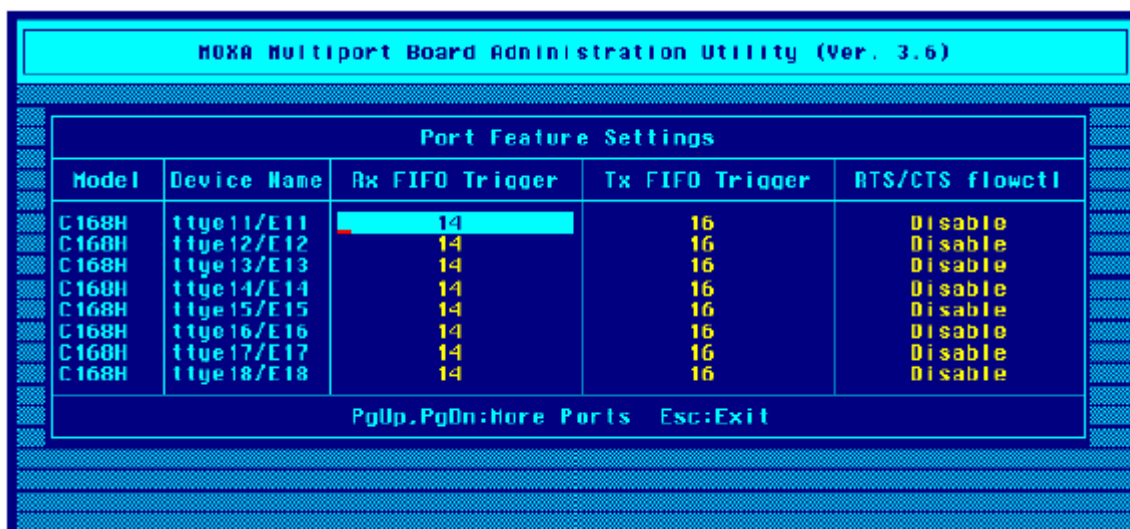
Label	High/Normal Speed Low Band	Normal Speed High Band	High Speed High Band
B50	50	57600	57600
B75	75	115200	115200
B110	110	110	230400
B134	134	134	460800
B150	150	150	921600
B200	200	200	200



B300	300	300	300
B600	600	600	600
B1200	1200	1200	1200
B1800	1800	1800	1800
B2400	2400	2400	2400
B4800	4800	4800	4800
B9600	9600	9600	9600
B19200	19200	19200	19200
B38400	38400	38400	38400

Обратите внимание: для получения интерактивной справки нажмите кнопку F1.

В поддиалоге [Port Feature Settings] есть три важных поля:



### Rx FIFO Trigger

Rx FIFO может принимать следующие значения: 1, 4, 8 или 14 байт; по умолчанию задается значение 14 байт.

### Tx FIFO S Trigger

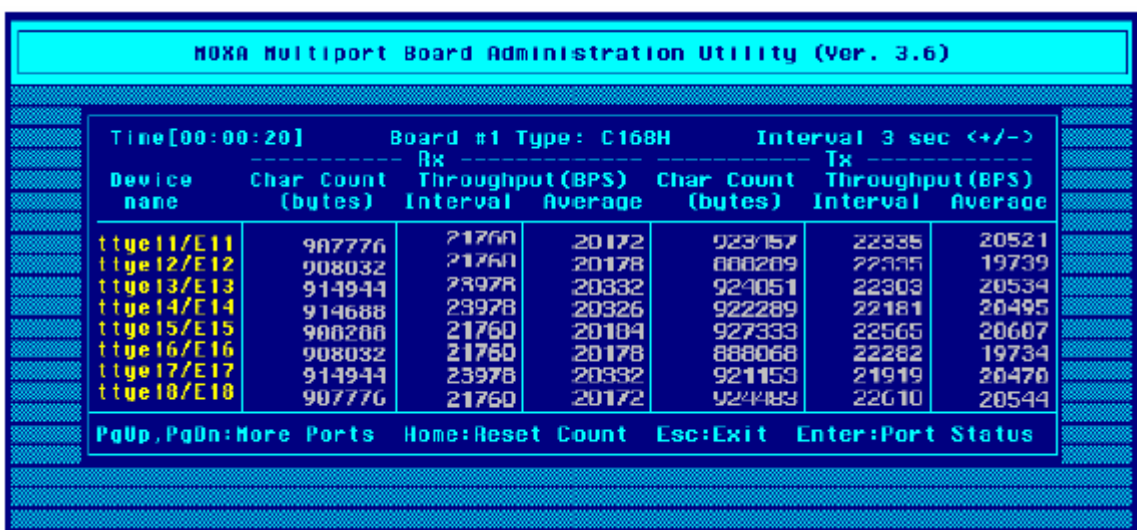
Допустимое значение Tx FIFO – от 1 до 16 байт. По умолчанию задается 16 байт.

## RTS/CTS Hardware Flow Control

Эта настройка делает поведение драйвера более гибким. Если установлено значение **"Disable"** (по умолчанию), то порту не будет нужен сигнал CTS, чтобы передать данные, и аппаратная функция RTS/CTS управления потоком данных в драйвере будет заблокирована. Напротив, если установлено значение **"Enable"**, порту будет необходим сигнал CTS, чтобы передавать данные, и функция RTS/CTS аппаратного управления потоком данных в драйвере соответственно реализуется.

## Port Monitoring

Эта утилита позволяет быстро контролировать активность портов MOXA. Вы сможете легко увидеть число принятых / переданных символов (Rx/Tx), начиная с момента начала контроля. Также сообщается Rx/Tx производительность в секунду за определенный интервал (например, за последние 5 секунд) и в среднем (с момента начала мониторинга). Вы можете сбрасывать все счётчики клавишей Home. Для того, чтобы изменить отображаемый интервал времени, используйте клавиши + и - (Плюс / минус).

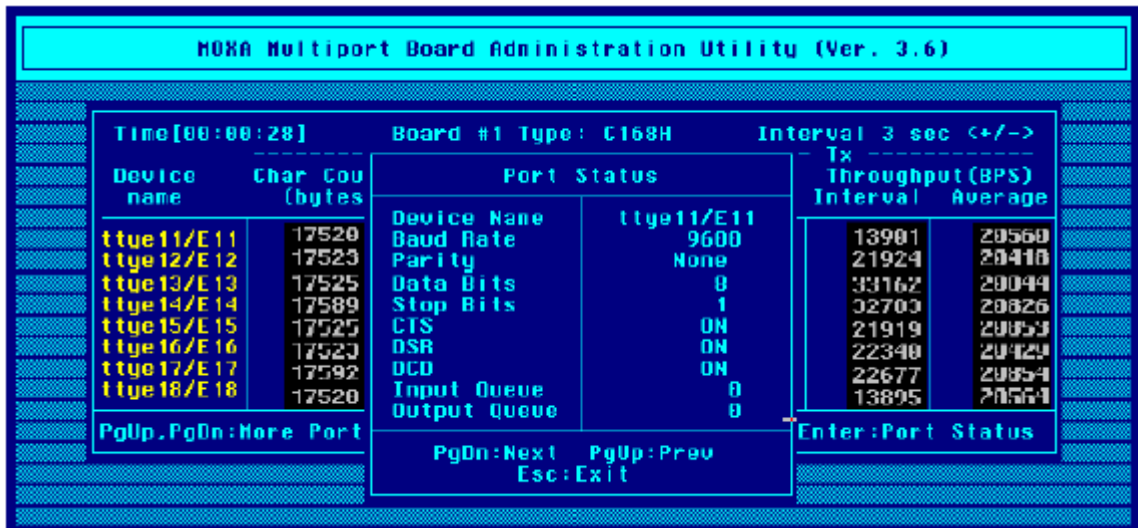


The screenshot displays the MOXA Multiport Board Administration Utility (Ver. 3.6) interface. It shows a table of port monitoring data for Board #1 (Type: C168H) with an interval of 3 seconds. The table lists device names, character counts, and throughput (BPS) for both Rx and Tx. At the bottom, there are control instructions: PgUp, PgDn: More Ports; Home: Reset Count; Esc: Exit; Enter: Port Status.

MOXA Multiport Board Administration Utility (Ver. 3.6)						
Time[00:00:20]		Board #1 Type: C168H		Interval 3 sec <+/->		
Device name	Char Count (bytes)	Rx		Tx		
		Interval	Throughput (BPS) Average	Char Count (bytes)	Interval	Throughput (BPS) Average
ttYe11/E11	907776	21760	20172	923157	22335	20521
ttYe12/E12	908032	21760	20178	886289	22335	19739
ttYe13/E13	914944	23978	20332	924051	22303	20534
ttYe14/E14	914688	23978	20326	922289	22181	20495
ttYe15/E15	906208	21760	20184	927333	22565	20607
ttYe16/E16	908032	21760	20178	888068	22282	19734
ttYe17/E17	914944	23978	20332	921153	21919	20470
ttYe18/E18	907776	21760	20172	924483	22610	20544

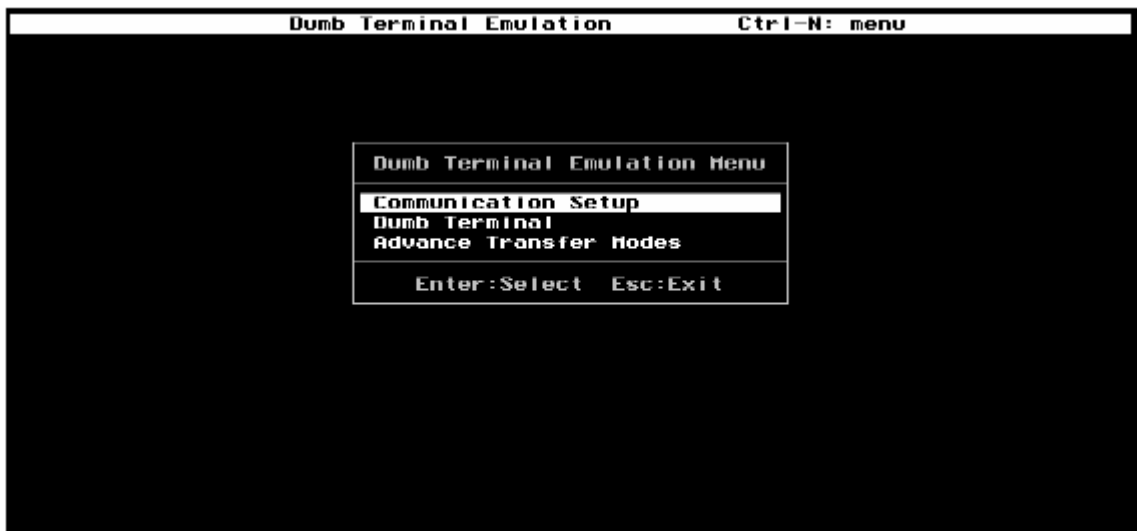
PgUp, PgDn: More Ports    Home: Reset Count    Esc: Exit    Enter: Port Status

Если Вы хотите увидеть параметры определенного порта, состояние сигнала и просмотреть очередь ввода-вывода, то наведите курсор на желаемый порт и нажмите Enter.

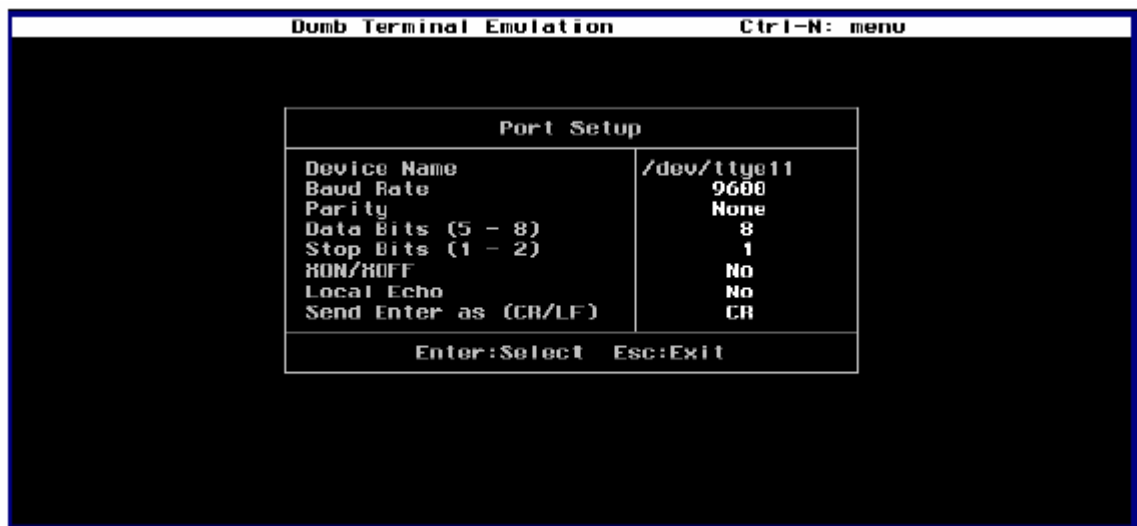


## Terminal Emulation

Эта утилита обеспечивает прием и отправку данных через любые порты TTY, в частности порты MOXA. Это весьма полезно для простых действий, например, посылка at команд в модем, подключенный к порту, или для входа в систему. Примечание: это простой эмулятор терминала ввода-вывода (dumb terminal), без обработки полноэкранной информации. Кроме того, доступна функция “data score”, с помощью которой можно пересылать файлы или пакеты.



1. Выберите пункт "**Communication Setup**" и нажмите Enter для введения коммуникационных параметров.



2. Чтобы запустить эмулятор терминала, выберите и введите пункт "**Dumb Terminal**", а чтобы осуществить пакетный или файловый обмен данными – пункт "**Advance Transfer Modes**".

## Driver Removal

Если Вы хотите удалить драйвер платы серии Smartio C168 и вернуться к предыдущей системной конфигурации, просто нажмите Enter на этом пункте меню и ответьте "Y" для подтверждения выбора.

Затем система будет перекомпилирована, что может занять некоторое время. Если Вы отвечаете "N", то не будет предпринято никаких действий.



## Подключение терминалов к портам MOXA

Чтобы разрешить использование портов MOXA с целью входа в систему, возьмем в качестве примера, терминал ttye11:

### SCO UNIX/OpenServer

```
# enable /dev/ttye11
```

### UNIX SVR4.2

1. Отредактируйте файл/etc/inittab (например, используйте редактор vi).
2. Измените параметры tty с " ma11:23:off:/etc/getty ttye11 9600 " на " ma11:23:respawn:/etc/getty ttye11 9600 "
3. # Init q

Или обратитесь к справочнику по системе UNIX для того, чтобы активизировать порт терминала.

## 4. Программное обеспечение

МОХА предлагает простые, но мощные библиотеки для программирования последовательного интерфейса и утилиты для отладки под Windows NT, Windows 95/98, UNIX и DOS; таким образом, программное обеспечение МОХА позволяет сэкономить много времени.

Следующие разделы подробно рассказывают об инсталляции библиотеки и утилит под некоторые платформы.

### Windows NT and Windows 95/98

**PComm**, профессиональное программное обеспечение для работы с последовательной связью, - это пакет программ для **Windows NT** и **Windows 95/98**, состоящий из библиотеки для простого программирования последовательного интерфейса на наиболее популярных языках, полезных утилит типа Diagnostic, Monitor и Terminal Emulation, наглядных примеров программ и подробных on-line документов.

Библиотека для программирования последовательного интерфейса полезна при настройке системы под работу с устройствами передачи данных, удаленным доступом, сбором данных или промышленным контролем в среде Windows NT и системе Windows 95/98, которая намного проще по сравнению с более сложной Windows Win32 COMM API.

### Инсталляция

Чтобы установить PCOMM, запустите с диска файл **Setup.exe**. Обратите внимание на то, что утилиты PCOMM Diagnostic и Monitor

работают только для плат MOXA и не поддерживают подобные платы других производителей. Необходимо, чтобы предварительно в систему была установлена плата MOXA и драйвера под Windows NT/95/98. Установка драйвера детально рассмотрена в главе «Установка драйвера»

## Библиотека функций PCOMM

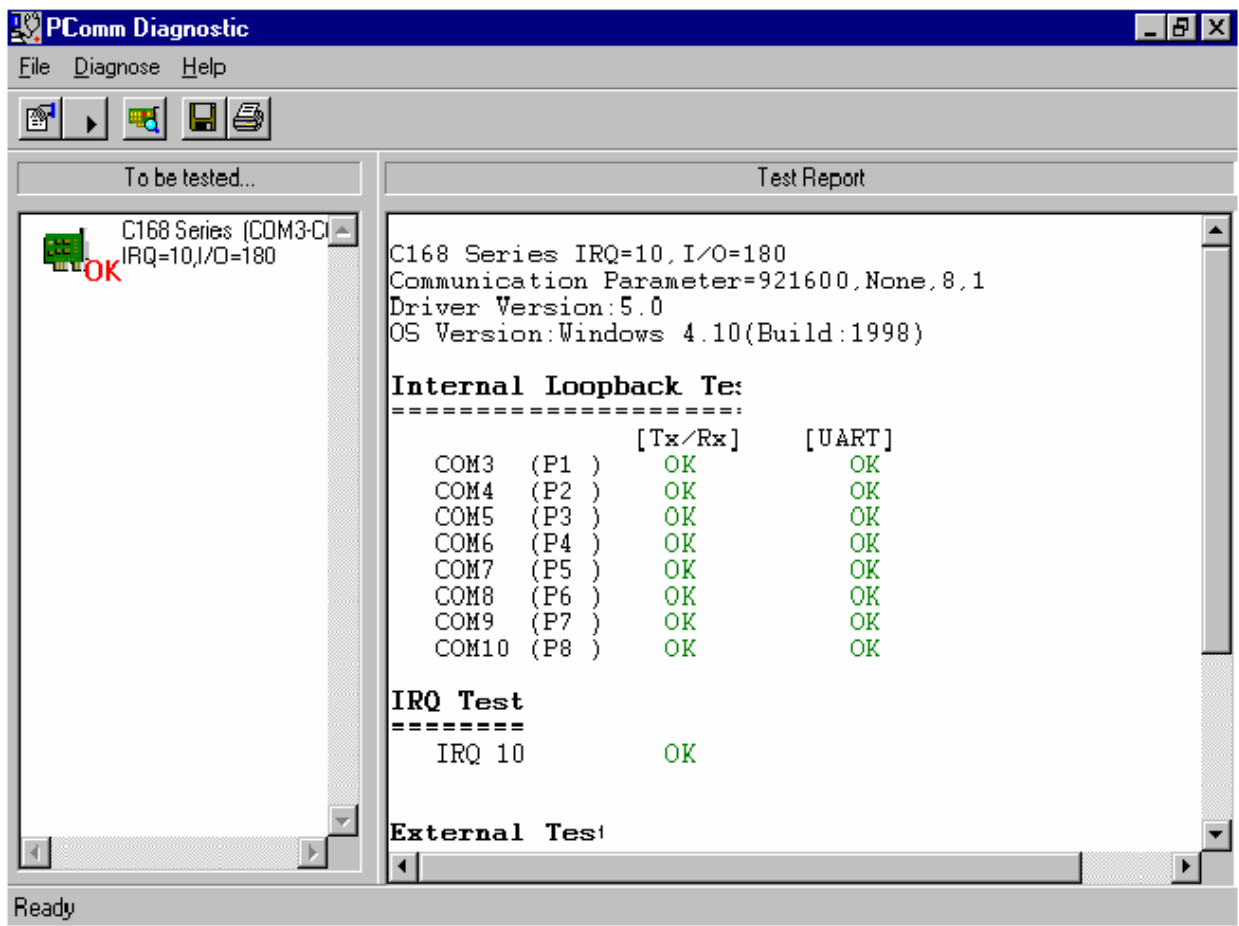
Библиотека для программирования должна помочь пользователям разрабатывать программы для последовательного интерфейса **любого СОМ порта**, совместимого с Microsoft Win32 API. Это может облегчить выполнение многозадачных и работающих с несколькими потоками данных программ, управляющих последовательной связью и, следовательно, позволяет существенно экономить время. Для изучения полного перечня функций библиотеки и примеров программ на Visual C++, Visual Basic и Delphi, обратитесь к справочному файлу и примерам программ в директории PCOMM.

## Утилиты

Далее следует краткое описание каждой утилиты. Более подробно, пожалуйста, см. **интерактивную справку** о запущенной утилите.

### Diagnostic (только для плат MOXA)

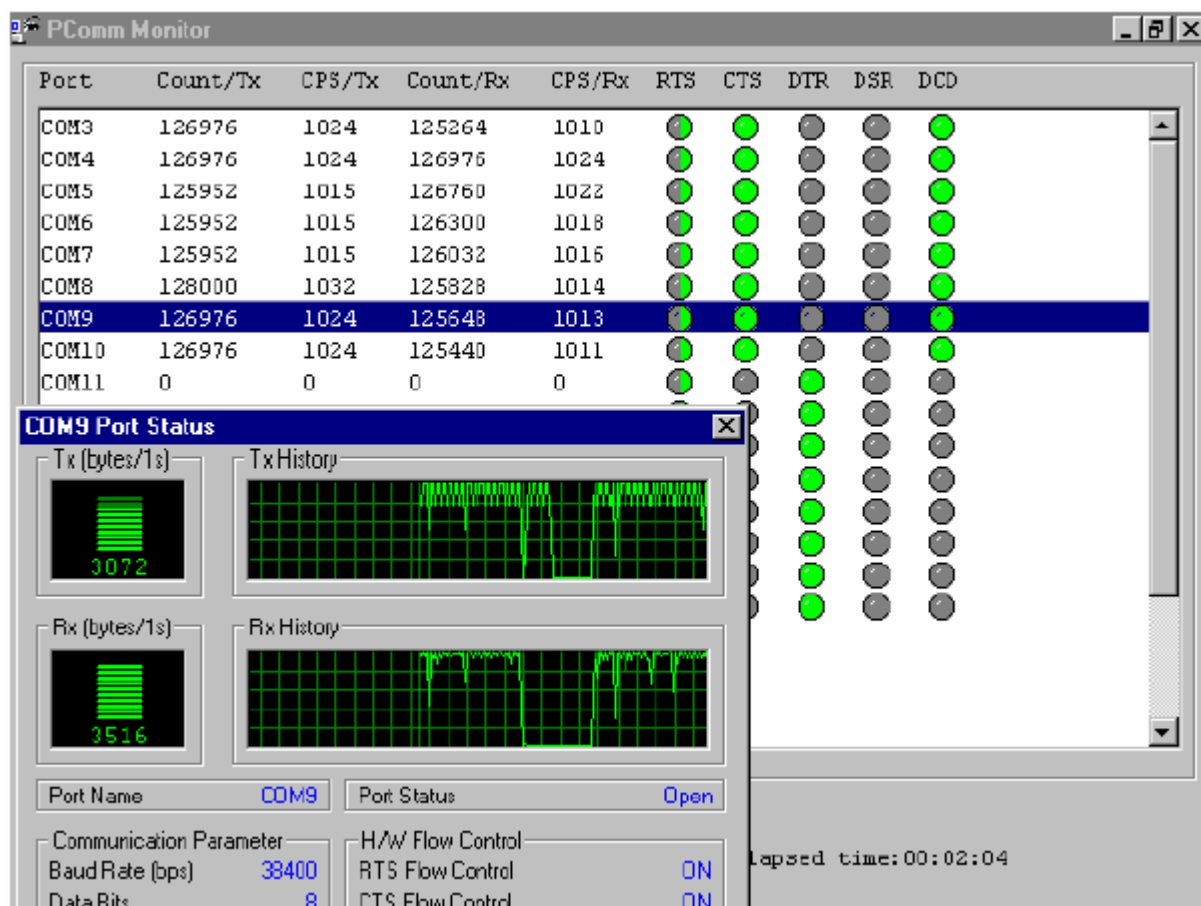
Удобная диагностическая программа обеспечивает внутреннее и внешнее тестирование (тест IRQ, UART, TxD/RxD, CTS/RTS, DTR/DSR, DTR/DCD и т.д.) для плат и портов MOXA, что позволяет проверить правильность функционирования оборудования и программного обеспечения.



## Monitor (только для плат MOXA под Windows NT)

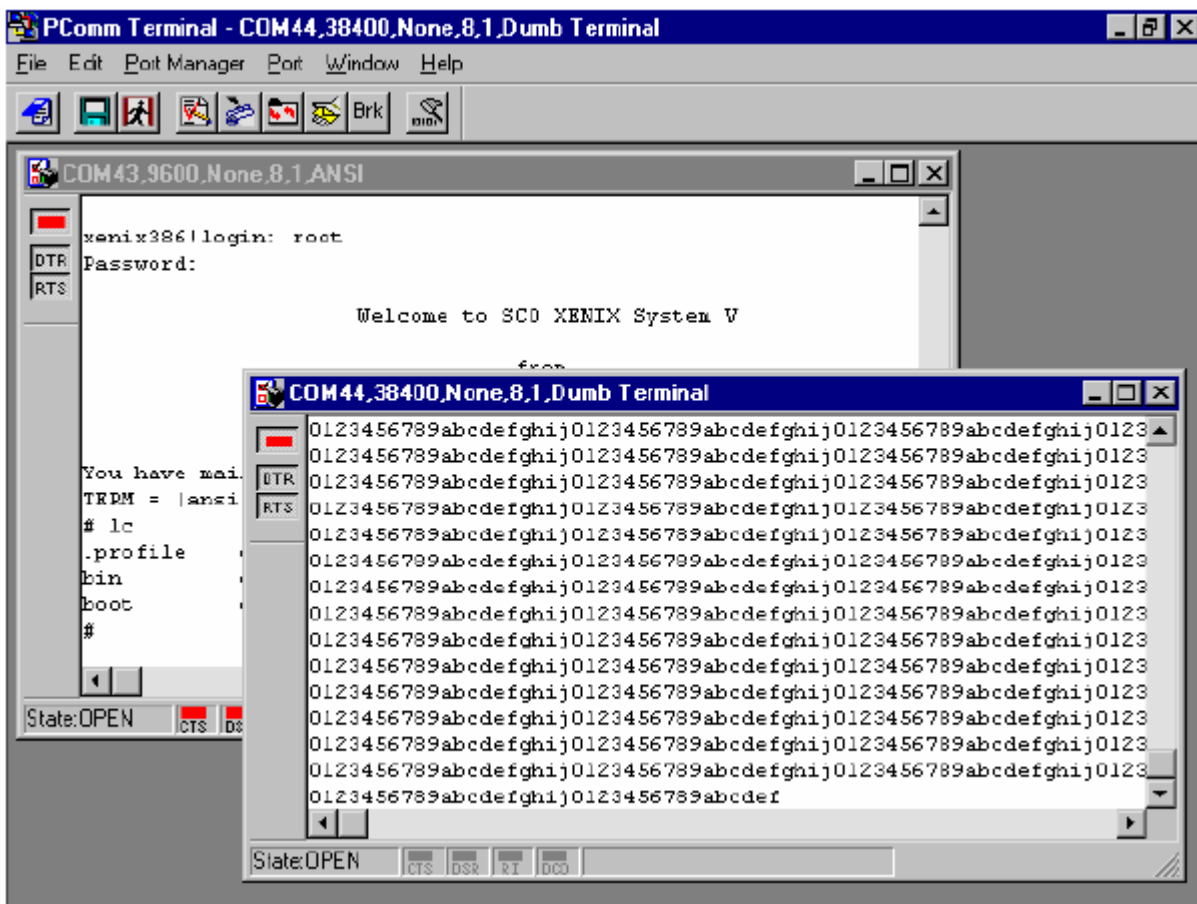
Данная программа мониторинга позволяет вам в любой момент времени наблюдать за используемыми COM портами MOXA, скоростью передачи / получения данных и состоянием линии связи, которые в реальном времени отражаются в окне утилиты. Кроме того, Вы можете нажать на изображение одного из портов и увидеть текущие параметры связи и состояние этого порта.





## Terminal Emulation

Эмулятор терминала имеет многооконный интерфейс и поддерживает терминалы типа VT100 и ANSI. Также вы можете передавать данные в интерактивном режиме, периодически посылать пакеты или пересылать какие-либо файлы, используя протоколы ASCII, XMODEM, YMODEM, ZMODEM и KERMIT.



## UNIX

### Программирование портов MOXA

К портам MOXA могут применяться команды, используемые для стандартных терминальных устройств UNIX, т.к. порты MOXA соответствуют стандарту tty UNIX. Используются такие системные команды как open(), ioctl(), read(), write(), close() и т.д. Пожалуйста, обратитесь к вашему справочнику программиста UNIX.

Тем не менее, это позволяет обеспечить только ограниченный набор функций, и таким образом не может удовлетворить запросов профессиональных программистов. Чтобы обеспечить выполнение специфических задач, MOXA поддерживает расширенные функции посредством команд ioctl(), в частности:

1. MIBUFED (= 0x401) Получить количество байт, находящихся во входном буфере.
2. MOBUFED (= 0x402) Получить количество байт, находящихся в выходном буфере.
3. MTCRTS (= 0x403) Управление выходным сигналом RTS.
4. MTCDSR (= 0x404) Управление выходным сигналом DSR.
5. MLOWATER (= 0x405) Установить уровень заполнения буфера вывода.
6. MSTATUS (= 0x407) Отображать состояние модемных линий (CTS/DSR/DCD).
7. MHWFLOW (= 0x40e) Включить / отключить аппаратное управление потоком данных.

Следующий раздел подробно описывает все команды.

## Расширенные команды UNIX ioctl ()

Далее следуют списки расширенных функций MOXA для SVR4.x и не-SVR4.x UNIX. Переменная `moxa_fd` - это дескриптор файла, возвращаемый функцией `open()` для конкретного порта MOXA. Например,

```
int moxa_fd;  
moxa_fd = open("/dev/tty11", O_RDWR);
```

### 1. MIBUFED

Эта функция сообщает, сколько байт поставлено в очередь во входном буфере в момент ее запуска.

**Синтаксис для SCO UNIX / OpenServer**

```
#define MIBUFED    0x401  
int               count; /* number of bytes queued in the buffer */  
ioctl(moxa_fd, MIBUFED, &count);
```

## Синтаксис для UNIX SVR4.2

```
#include <sys/stropts.h>
#include <sys/sysmacros.h>
#define MIBUFED 0x401
struct strioctl ioc;
int count; /* number of bytes queued in the buffer */

ioc.ic_cmd = MIBUFED;
ioc.ic_timeout = 0;
ioc.ic_len = sizeof(int);
ioc.ic_dp = (char *)&count;
ioctl(moxa_fd, I_STR, &ioc);
```

**Обратите внимание:** из-за особенности работы потокового драйвера, эта функция возвращает число байт, которые буферизированы на плате МОХА, не включая данные, буферизированные потоковым драйвером. В данном случае это число используется только в справочных целях. Например, возвращаемое число может быть всегда равно нулю, но при этом имеются данные, буферизированные потоковым драйвером.

## 2. MOBUFED

Эта функция возвращает расчетное количество байт, поставленных в очередь в буфере вывода в момент запуска функции.

### Синтаксис для SCO UNIX / OpenServer

```
#define MOBUFED 0x402
int count; /* number of bytes queued in the output buffer */

ioctl(moxa_fd, MOBUFED, &count);
```

## Синтаксис для UNIX SVR4.2

```
#include <sys/stropts.h>
#include <sys/sysmacros.h>
#define MOBUFED 0x402
```

```

struct strioctl      ioc;
int                  count; /* number of bytes queued in the o utput buffer */

ioc.ic_cmd = MOBUFED;
ioc.ic_timeout = 0;
ioc.ic_len = sizeof(int);
ioc.ic_dp = (char *)&count;
ioctl(moxa_fd, I_STR, &ioc);

```

**Обратите внимание:** См. MIBUFED о влиянии потокового драйвера.

### 3. MTCRTS

Эта функция правильно работает только тогда, когда аппаратное управление потоком данных выключено (см. MHWFLOW), и используется, чтобы включать или выключать сигнал RTS.

#### Синтаксис для SCO UNIX / OpenServer

```

#define MTCRTS      0x403
#define TurnON      1
#define TurnOFF     0

ioctl(moxa_fd, MTCRTS, TurnON);
ioctl(moxa_fd, MTCRTS, TurnOFF);

```

#### Синтаксис для UNIX SVR4.2

```

#include          <sys/stropts.h>
#include          <sys/sysmacros.h>
#define MTCRTS   0x403
#define TurnON   1
#define TurnOFF  0
struct strioctl ioc;
int             setting;

setting = TurnON /* or TurnOFF */;
ioc.ic_cmd = MTCRTS;
ioc.ic_timeout = 0;
ioc.ic_len = sizeof(int);
ioc.ic_dp = (char *)&setting;
ioctl(moxa_fd, I_STR, &ioc);

```

## 4. MTCDDTR

Эта функция правильно работает только тогда, когда аппаратное управление потоком данных выключено (см. MHWFLOW), и используется для того, чтобы включать или выключать сигнал DTR.

### Синтаксис для SCO UNIX / OpenServer

```
#define MTCDDTR      0x404
#define TurnON      1
#define TurnOFF     0
ioctl(moxa_fd, MTCDDTR, TurnON);
ioctl(moxa_fd, MTCDDTR, TurnOFF);
```

### Синтаксис для UNIX SVR4.2

```
#include <sys/stropts.h>
#include <sys/sysmacros.h>
#define MTCDDTR      0x404
#define TurnON      1
#define TurnOFF     0
struct striocctl    ioc;
int                 setting;

setting = TurnON /* or TurnOFF */;
ioc.ic_cmd = MTCDDTR;
ioc.ic_timeout = 0;
ioc.ic_len = sizeof(int);
ioc.ic_dp = (char *)&setting;
ioctl(moxa_fd, I_STR, &ioc);
```

## 5. MLOWATER

Иногда приложение не способно записывать поступающие данные в буфер вывода из-за его заполнения. Приложение должно ждать до момента, когда в буфере вывода освободится достаточно места. Критерий, по которому определяется момент, когда приложению сообщается что в буфере вывода достаточно свободного места, называется уровнем заполнения (low water level). Соответственно,

буфер вывода продолжит получать данные только, когда уровень “low water” достигнут. Если значение “low water” относительно мало, Вы можете увидеть, что буфер вывода опустошается прежде, чем вы записываете в него другой блок данных (это вполне возможно, потому что UNIX - многозадачная среда с разделением времени). Результатом является прерывистая передача данных. В приложениях, для которых важны временные параметры, например факсимильная передача, прерванный поток данных может нарушить связь.

По умолчанию “low water” имеет значение 512 байт. Вы можете увеличить эту величину, но лучше не превышать половину объема буфера вывода. Буфер вывода каждого порта - 32КБ.

### **Синтаксис для SCO UNIX / OpenServer**

```
#define MLOWATER 0x405
int lowater; /* low water value of output buffer (default = 512 bytes) */

ioctl(moxa_fd, MLOWATER, lowater);
```

### **Синтаксис для UNIX SVR4.2**

```
#include <sys/stropts.h>
#include <sys/sysmacros.h>
#define MLOWATER 0x405
struct strioctl ioc;
int lowater; /* low water value of output buffer (default = 512 bytes) */
ioc.ic_cmd = MLOWATER;
ioc.ic_timeout = 0;
ioc.ic_len = sizeof(int);
ioc.ic_dp = (char *)&lowater;
ioctl(moxa_fd, I_STR, &ioc);
```

## **6. MSTATUS**

Эта функция информирует о состоянии модемных линий RS-232 (CTS/DSR/DCD).

### **Синтаксис для SCO UNIX / OpenServer**

```

#define MSTATUS      0x407
int    status;      /* status = RS-232 line status */
                          /* bit0 CTS (1:on, 0:off) */
                          /* bit1 DSR (1:on, 0:off) */
                          /* bit2 DCD (1:on, 0:off) */

ioctl(moxa_fd, MSTATUS, &status);

```

## Синтаксис для UNIX SVR4.2

```

#define MSTATUS      0x407
#include             <sys/stropts.h>
#include             <sys/sysmacros.h>

struct strioctl     ioc;
int    status;      /* status = RS-232 line status */
                          /* bit0 CTS (1:on, 0:off) */
                          /* bit1 DSR (1:on, 0:off) */
                          /* bit2 DCD (1:on, 0:off) */

ioc.ic_cmd = MSTATUS;
ioc.ic_timeout = 0;
ioc.ic_len = sizeof(int);
ioc.ic_dp = (char *)&status;
ioctl(moxa_fd, I_STR, &ioc);

```

## 7. MHWFLOW

Эта функция используется, чтобы включить / отключить аппаратное управление потоком данных. Первая функция `open()` установит/очистит биты аппаратного управления потоком в зависимости от параметра настройки **[RTS/CTS Hardware Flow Control]** в `moxaadm`. Однако, если вы решите по желанию управлять сигналом DTR или RTS, то бит управления потоком данных RTS должен быть очищен (`HWFlowControlOff`), чтобы осуществлять контроль за сигналами DTR или RTS через функцию `MTCRTS` или `MTCDTR`. `MTCRTS` и `MTCDTR` будут работать только после того, как управление потоком данных с помощью RTS `MHWFLOW` будет выключено.



## Синтаксис для SCO UNIX / OpenServer

```
#define MHWFLOW          0x40e
#define HWFlowControlOff 0x00
#define CTSFlowControlBitOn 0x01
#define RTSFlowControlBitOn 0x02
#define HWFlowControlOn  0x03

ioctl(moxa_fd, MHWFLOW, CTSFlowControlBitOn);
ioctl(moxa_fd, MHWFLOW, RTSFlowControlBitOn);
ioctl(moxa_fd, MHWFLOW, CTSFlowControlBitOn | RTSFlowControlBitOn);
```

## Синтаксис для UNIX SVR4.2

```
#include <sys/stropts.h>
#include <sys/sysmacros.h>

#define MHWFLOW          0x40e
#define HWFlowControlOff 0x00
#define CTSFlowControlBitOn 0x01
#define RTSFlowControlBitOn 0x02
#define HWFlowControlOn  0x03
struct striocctl          ioc;
int                        setting;
setting = CTSFlowControlbitOn;
ioc.ic_cmd = MHWFLOW;
ioc.ic_timeout = 0;
ioc.ic_len = sizeof(int);
ioc.ic_dp = (char *)&setting;
ioctl(moxa_fd, I_STR, &ioc);
```

## УТИЛИТЫ

Вы можете использовать административную утилиту **moxaadm** для того, чтобы контролировать **активность порта** и осуществлять **эмуляцию терминала**. Подробности приводятся ниже.

## Port Monitoring

Эта утилита позволяет быстро контролировать активность портов MOXA. Вы сможете легко увидеть число принятых / переданных символов (Rx/Tx), начиная с момента начала контроля. Также сообщается Rx/Tx производительность в секунду за определенный интервал (например, за последние 5 секунд) и в среднем (с момента начала мониторинга). Вы можете сбрасывать все счётчики клавишей Home. Для того, чтобы изменить отображаемый интервал времени, используйте клавиши + и - (Плюс / минус).

MOXA Multiport Board Administration Utility (Ver. 3.6)						
Time[00:00:20]		Board #1 Type: C168H		Interval 3 sec <+/->		
Device name	Char Count (bytes)	Rx		Tx		
		Interval	Throughput (BPS) Average	Char Count (bytes)	Interval	Throughput (BPS) Average
ttue11/E11	907776	21760	20172	923157	22335	20521
ttue12/E12	908032	21760	20178	888289	22335	19739
ttue13/E13	914944	23978	20332	924051	22303	20534
ttue14/E14	914688	23978	20326	922289	22181	20495
ttue15/E15	900200	21760	20104	927333	22565	20607
ttue16/E16	908032	21760	20178	888068	22282	19734
ttue17/E17	914944	23978	20332	921153	21919	20470
ttue18/E18	907776	21760	20172	924483	22610	20544

PgUp, PgDn: More Ports    Home: Reset Count    Esc: Exit    Enter: Port Status

Если Вы хотите увидеть параметры определенного порта, состояние сигнала и просмотреть очередь ввода-вывода, то наведите курсор на желаемый порт и нажмите Enter.

MOXA Multiport Board Administration Utility (Ver. 3.6)						
Time[00:00:28]		Board #1 Type: C168H		Interval 3 sec <+/->		
Device name	Char Count (bytes)	Port Status		Tx		
		Device Name	ttue11/E11	Interval	Throughput (BPS) Average	
ttue11/E11	17520	Baud Rate	9600	13901	20560	
ttue12/E12	17523	Parity	None	21924	20410	
ttue13/E13	17525	Data Bits	8	33162	20044	
ttue14/E14	17589	Stop Bits	1	32703	20626	
ttue15/E15	17525	CIS	ON	21919	20053	
ttue16/E16	17523	DSR	ON	22340	20424	
ttue17/E17	17592	DCD	ON	22677	20554	
ttue18/E18	17520	Input Queue	0	13895	20561	
		Output Queue	0			

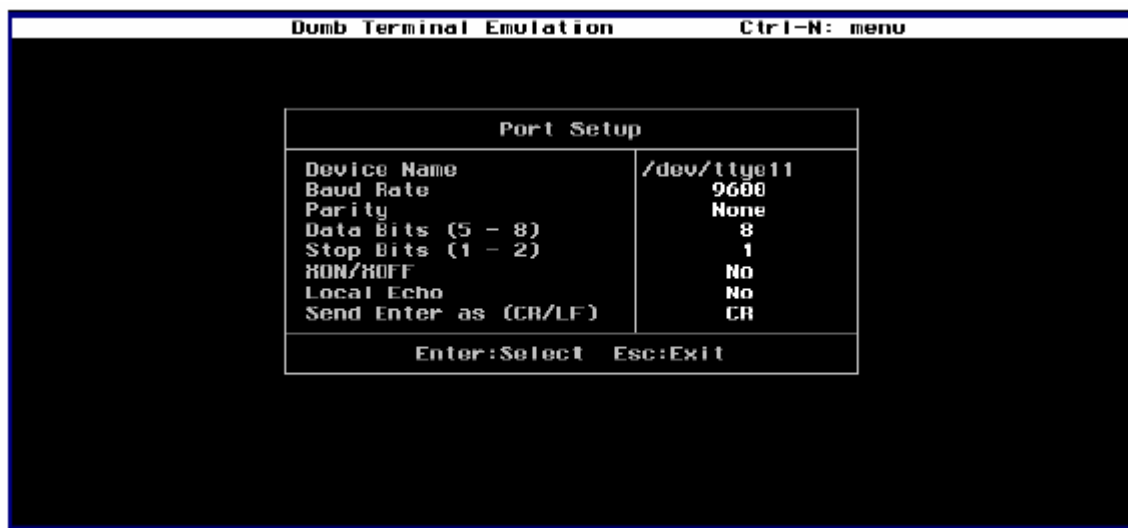
PgUp, PgDn: More Port    Enter: Port Status

PgDn: Next    PgUp: Prev    Esc: Exit

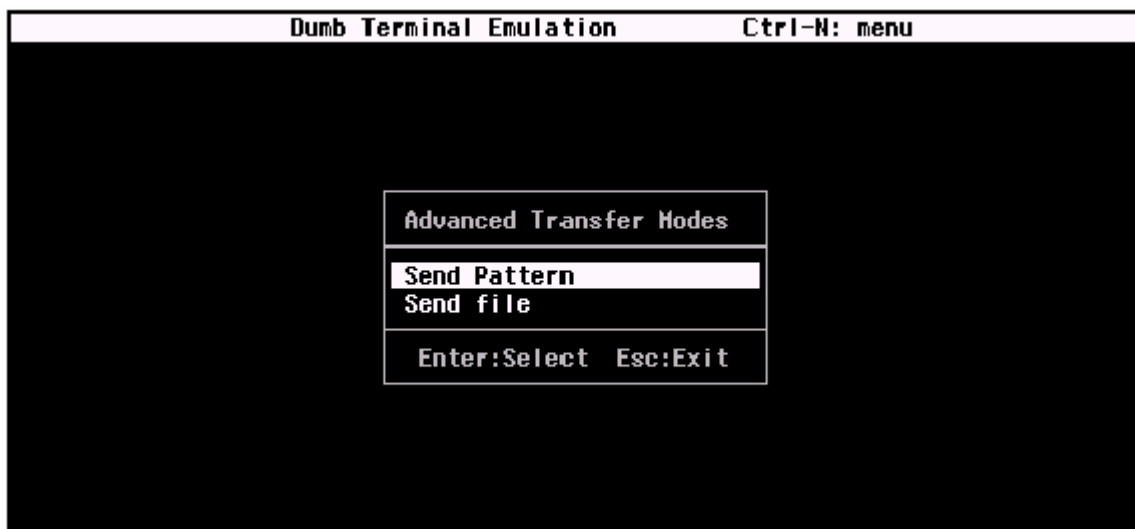
## Terminal Emulator

Эта утилита обеспечивает прием и отправку данных через любые порты TTY, в частности порты MOXA. Это весьма полезно для простых действий, например, посылка at команд в модем подключенный к порту или для входа в систему. Это простой эмулятор терминала ввода-вывода (dumb terminal), без обработки полноэкранный информации. Кроме того, доступна функция "data score", с помощью которой можно пересылать файлы или пакеты.

1. Выберите пункт **"Communication Setup"** и нажмите Enter для введения коммуникационных параметров.



2. Выберите поле **"Dumb Terminal"** и нажмите Enter, чтобы запустить эмуляцию терминала или поле **"Advanced Transfer Modes"**, чтобы осуществить передачу пакетов или файлов с помощью протоколов типа ZMODEM.



## DOS

### Инсталляция

Библиотека API-232 - профессиональное программное обеспечение, разработанное под операционную систему DOS. Она устанавливается автоматически наряду с драйверами MOXA под DOS. Детальное описание процесса инсталляции дано в главе "Инсталляция драйвера".

### Библиотека DOS API-232

Библиотека DOS API -232 поддерживает такие языки программирования как Microsoft C, Turbo C, Macro Assembler, QuickBasic, Turbo Pascal, Clipper и т.д. Примеры программ для

каждого поддерживаемого языка прилагаются; они размещены на диске в поддиректории ...\**EXAMPLE**\language.

Кроме того, только для языка C под DOS, имеется также библиотека управления модемом и передачи файлов, поддерживающая Hayes совместимое управление модемом, и протоколы ASCII, KERMIT, XMODEM, YMODEM ZMODEM.

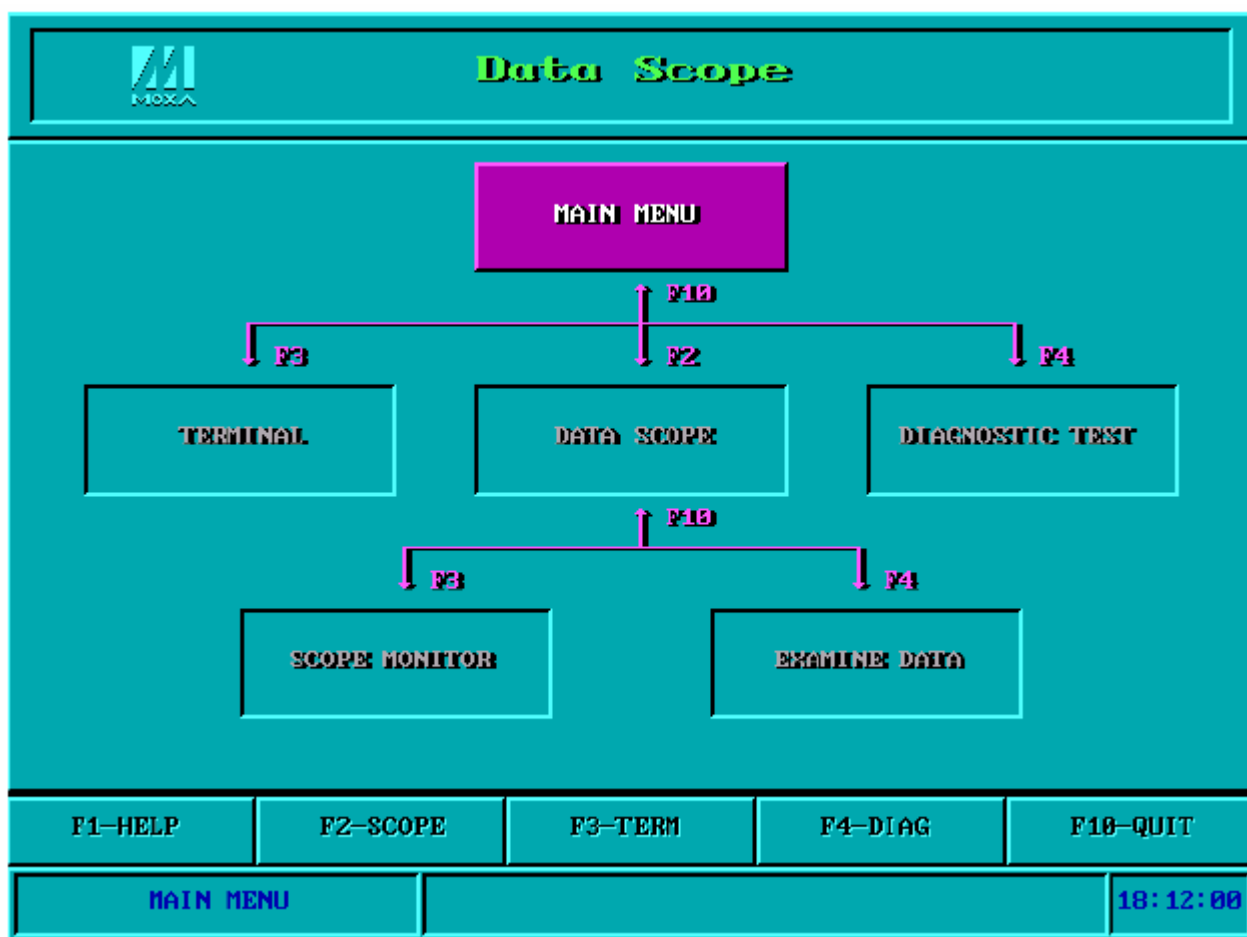
Более полное описание функций API-232 содержится в файле **API-232.TXT**, расположенном в директории API-232.

## **Утилиты**

Имеются две утилиты, работающие под DOS: Data Scope и Diagnose. Подробности описаны ниже.

### **Data Scope**

Data Scope, \**BIN\SCOPE.EXE**, является набором утилит, которые помогают пользователю решать возникающие в системе проблемы и отлаживать последовательную связь.



В утилите Data Scope имеются три основные функции:

1. Функция **Data Scope** осуществляет прозрачный контроль линий последовательной связи и позволяет сохранять данные на диске для проведения последующего анализа.
2. Функция **TTY terminal emulation** позволяет следить за состоянием сигнала и передавать данные в интерактивном режиме или файлы, используя протоколы ASCII, XMODEM, YMODEM, ZMODEM и KERMIT.
3. Функция **Diagnostic test** обеспечивает проверку линии между двумя портами MOXA, соединенных внешним кабелем.

Для более детального ознакомления с данным вопросом, пожалуйста, см. **интерактивную справку**, запустив BIN\SCOPE.EXE.

## Diagnose

Diagnose, \BIN\DIAGNOSE.EXE, - это утилита, позволяющая пользователю самостоятельно диагностировать состояние каждого порта выбранной платы. Более подробно см. интерактивную справку.

Перед выполнением этой процедуры, пожалуйста, заранее удалите драйвер MOXA из системы, выполнив "Mx-drv /Q", если драйвер MOXA работает в резидентном режиме.



## Программирование RS-485 для Opt8J

Если Вы намереваетесь налаживать RS-485 связь с помощью Opt8J, пожалуйста, следуйте руководству по программированию RS-485, приведенному ниже, а также ознакомьтесь с нюансами, описанными в главе "Соединительные модули (Opt8x) и распайка разъемов".

Opt8J поддерживает **только 2-проводную полудуплексную RS-485 связь**. Линии **Data +/-** обслуживают оба направления потока данных, и передачу, и прием, в зависимости от сигнала RTS.

Переключатель каждого порта нужно установить в положение **On**. Если **имеется сигнал RTS**, то порт будет передавать данные, а если его **нет** - тогда будет принимать их.

Схема **RTS** подходит для любой системы, включая Windows NT и Windows 95/98, DOS и UNIX, приложения которых разрешают контроль RTS из приложений.

Ниже приводятся два способа управления RS-485 2-проводной передачей.

### Способ 1

Обычно для RS-485 2-проводной передачи используется следующая модель.

```
sio_SetWriteTimeouts(port, 0); /* Устанавливается sio_write()
                                в режим работы с блоками*/
sio_RTS(port, 1);              /* Включается сигнал RTS.
                                Порт RS-485 готов к передаче
                                данных*/
sio_write(port, buff, 10);     /* 10 байт записываются в
                                "buff". Функции блокируются,
                                пока не передадутся все
                                символы*/
sio_RTS(port, 0);             /* Выключается сигнал RTS.
                                Порт RS-485 готов к приему
                                данных*/
sio_read(port, buff, 10);     /* Считываются 10 байт */
```

### Способ 2

В библиотеках PCOMM и API-232 имеется специализированная функция RS-485. В ней вышеупомянутые функции интегрированы в одно действие.

```
sio_putb_x(port, buff, tick ); /* 1.Включается RTS; готовность
                                к передаче данных
                                2.Передача данных.
```



**3.Таймаут.**

**4.Выключается сигнал RTS;  
готовность к приему  
данных.\*!**

Для получения дополнительной информации об этих функциях, пожалуйста, обратитесь к файлу интерактивной справки библиотеки PCOMM для Windows NT и Windows 95/98 или соответственно API-232.txt для DOS.

## 5. Соединительные модули (Opt8x) и распайка разъемов

В последовательной передаче данных используются термины:

- **DTE** - оборудование типа COM1/2 персонального компьютера, последовательный принтер и терминал.
- **DCE** - оборудование типа модема.

### Распайка разъемов RS-232 Opt8A/B/C/D/S

Для Smartio C168 разработаны следующие 8-портовые RS-232 соединительные модули/кабели:

**Opt8A:** RS-232 DB25 8-портовый соединительный модуль (гнезда)

**Opt8B:** RS-232 DB25 8-портовый соединительный модуль (штекеры)

**Opt8C:** кабель с 8 RS-232 DB25 разъемами

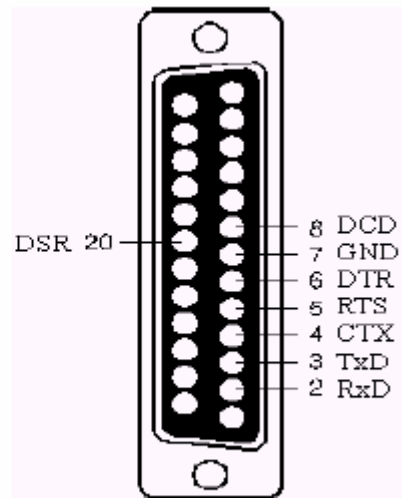
**Opt8D:** кабель с 8 RS-232 DB9 разъемами

**Opt8S:** RS-232 DB25 8-портовый соединительный модуль (гнезда) с защитой от выбросов напряжения

Далее следует описание контактов в разъеме для разных устройств:

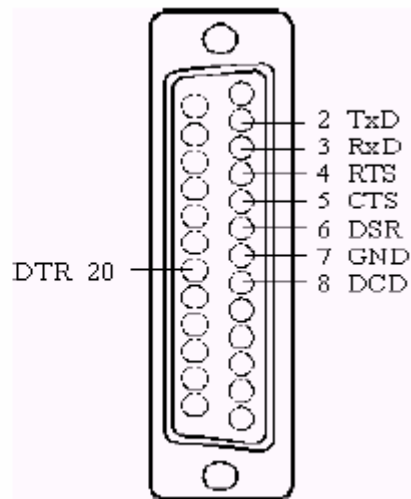
**Opt8A/S  
(DCE, DB25 Female)**

2	RxD
3	TxD
4	CTS
5	RTS
6	DTR
7	GND
8	DCD
20	DSR



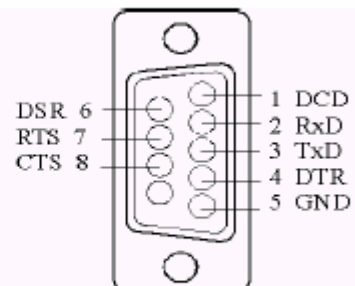
**Opt8B/C  
(DTE, DB25 Male)**

2	TxD
3	RxD
4	RTS
5	CTS
6	DSR
7	GND
8	DCD
20	DTR



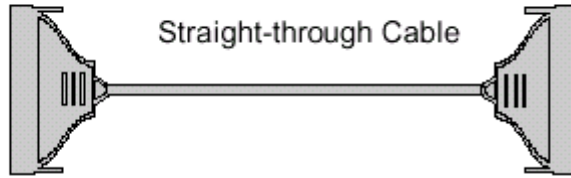
**Opt8D  
(DTE, DB9 Male)**

1	DCD
2	RxD
3	TxD
4	DTR
5	GND
6	DSR
7	RTS
8	CTS



**Тип 1: Соединение Smartio C168 с устройством DTE.**

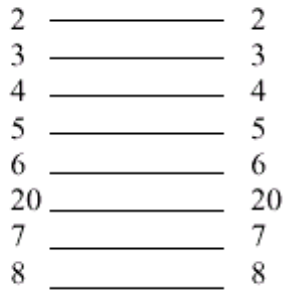
Opt8A/S



PC COM2 port, Serial Printer, Terminal, or any DTE Device

**Opt8A/S**  
**DB25 Female**

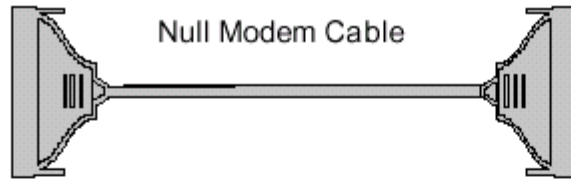
RxD  
TxD  
CTS  
RTS  
DTR  
DSR  
GND  
DCD



**DTE Device**  
**DB25 Male**

TxD  
RxD  
RTS  
CTS  
DSR  
DTR  
GND  
DCD

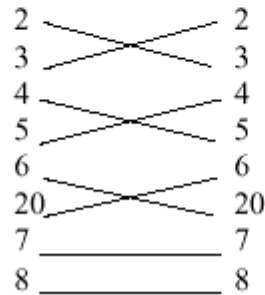
Opt8B/C



PC COM2 port, Serial Printer, Terminal, or any DTE Device

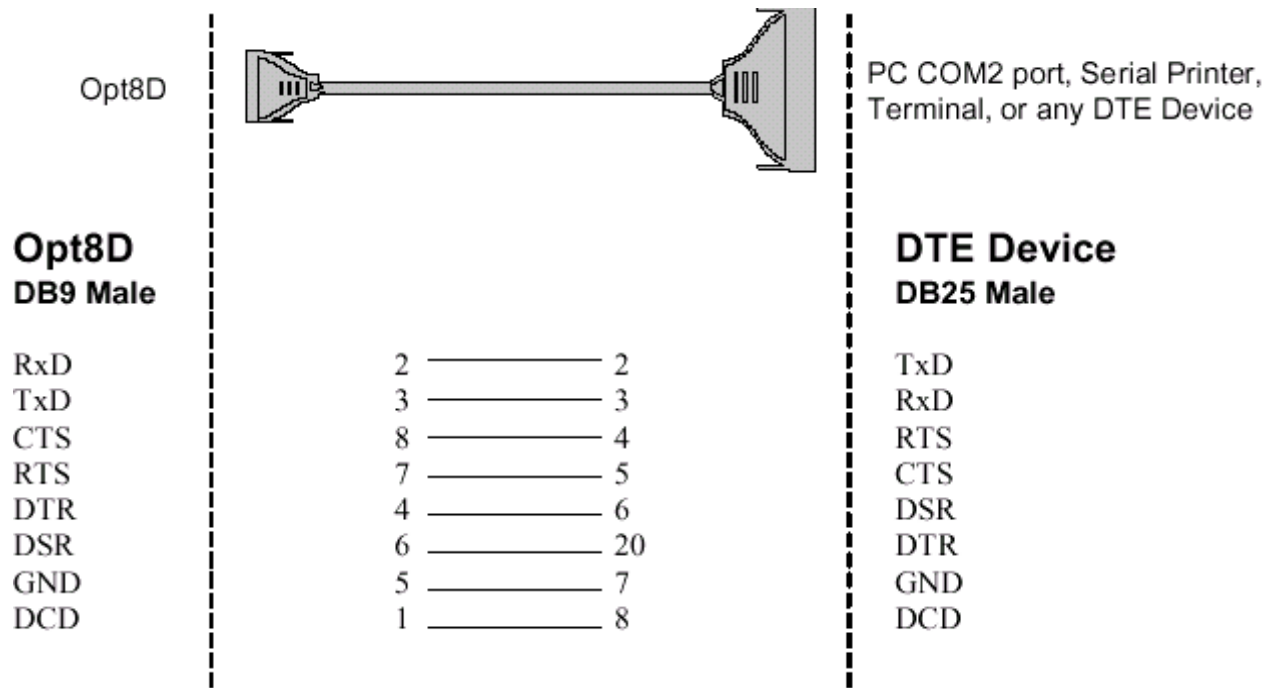
**Opt8B/C**  
**DB25 Male**

TxD  
RxD  
RTS  
CTS  
DSR  
DTR  
GND  
DCD

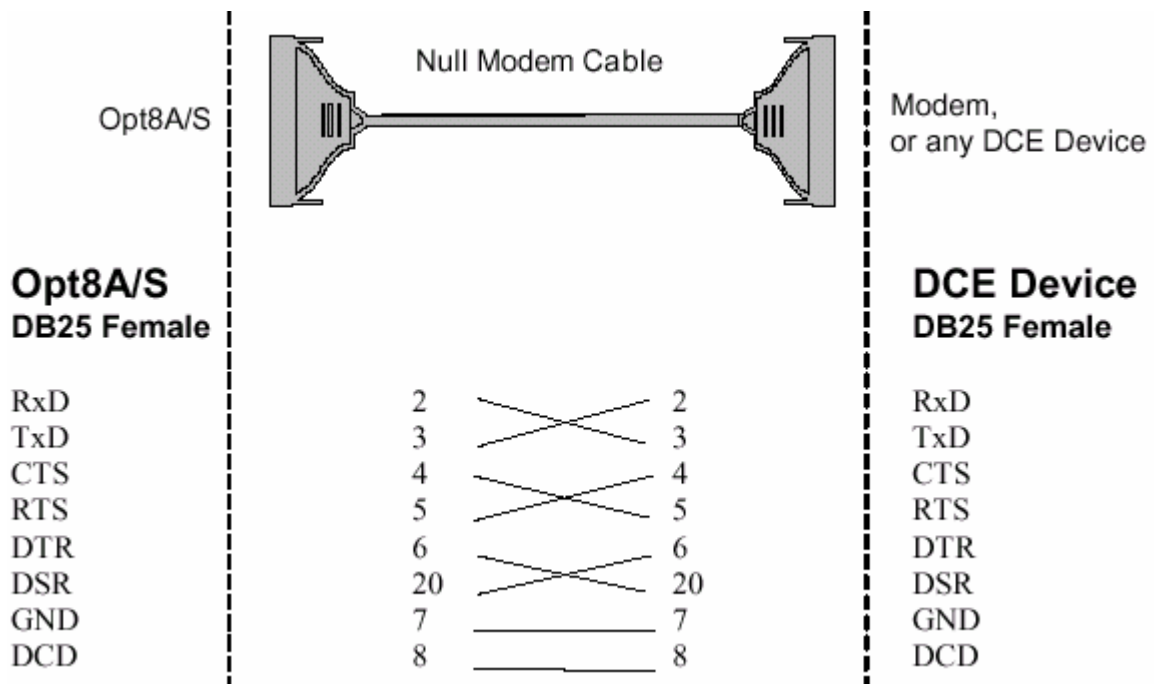


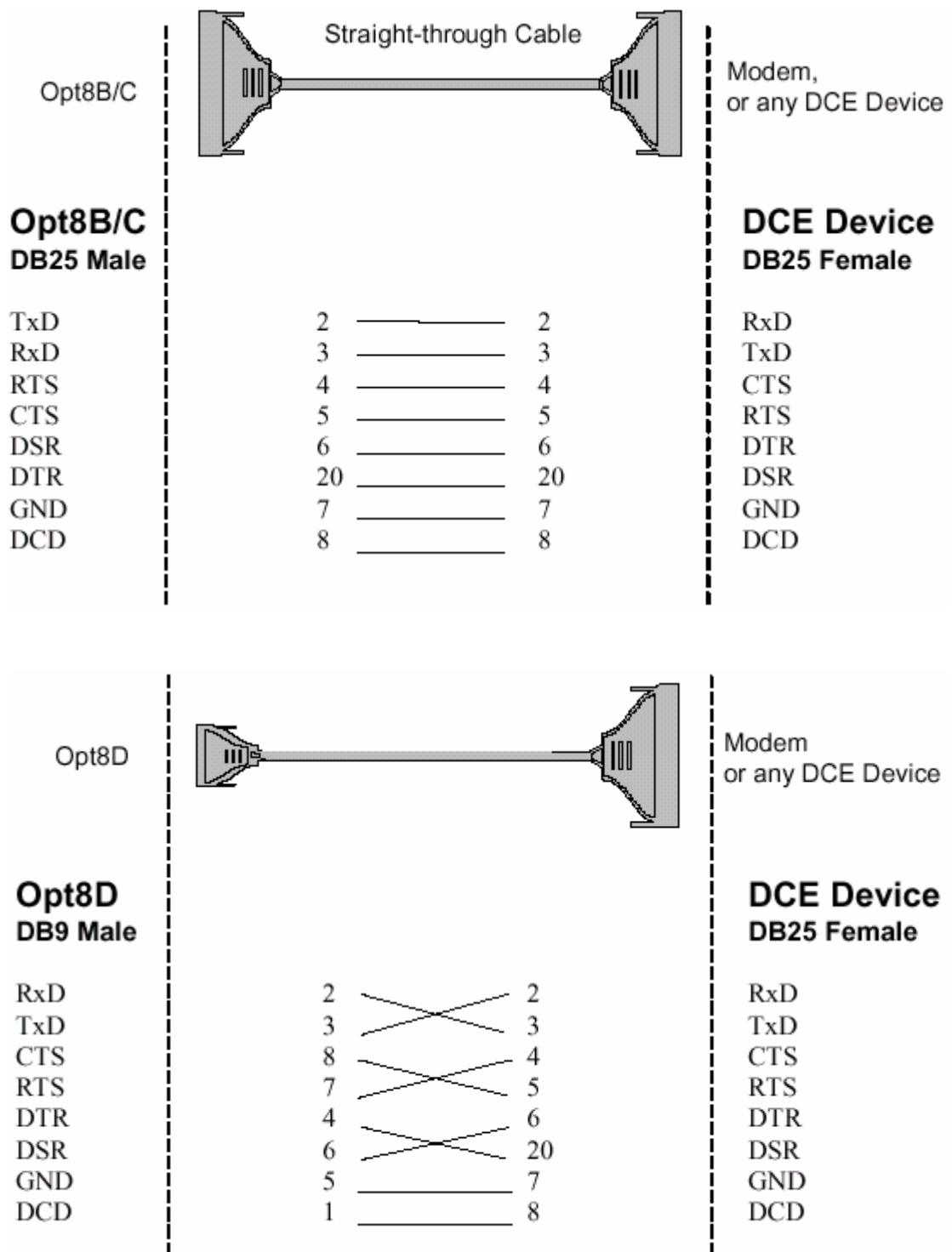
**DTE Device**  
**DB25 Male**

TxD  
RxD  
RTS  
CTS  
DSR  
DTR  
GND  
DCD



Тип 2: Соединение Smartio C168 с устройством DCE.

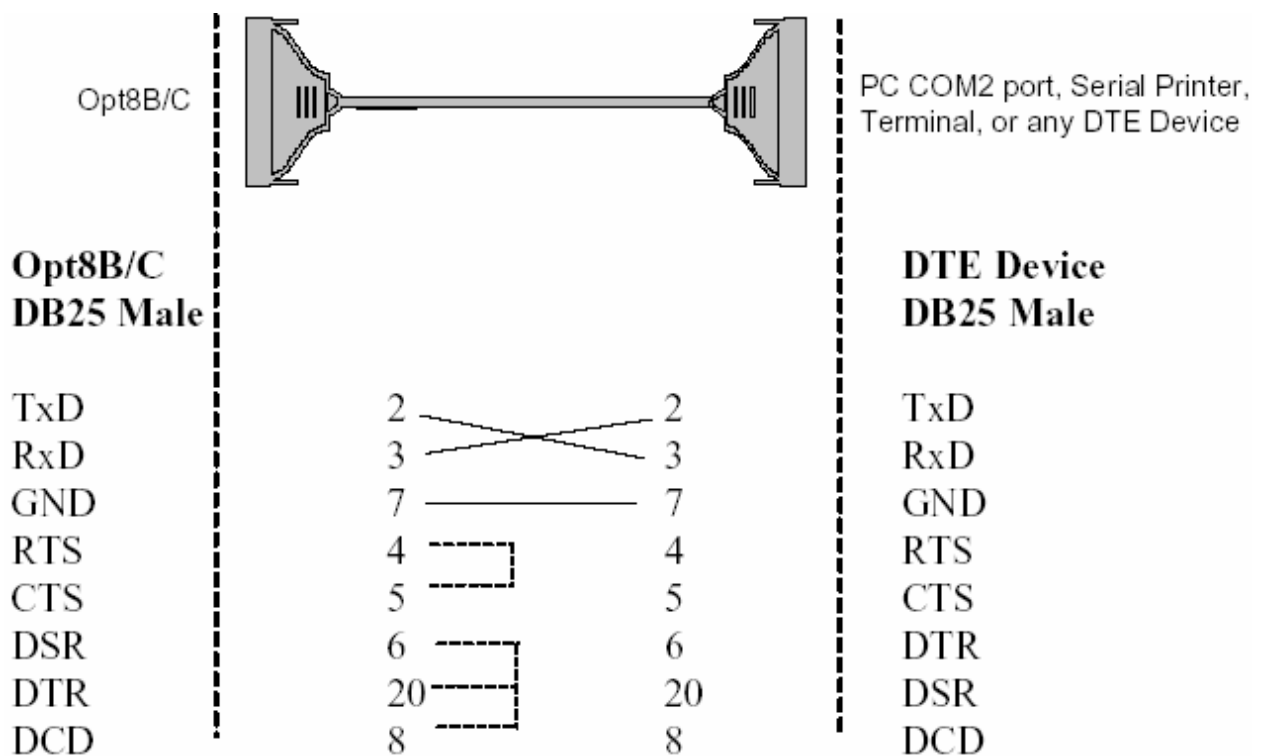
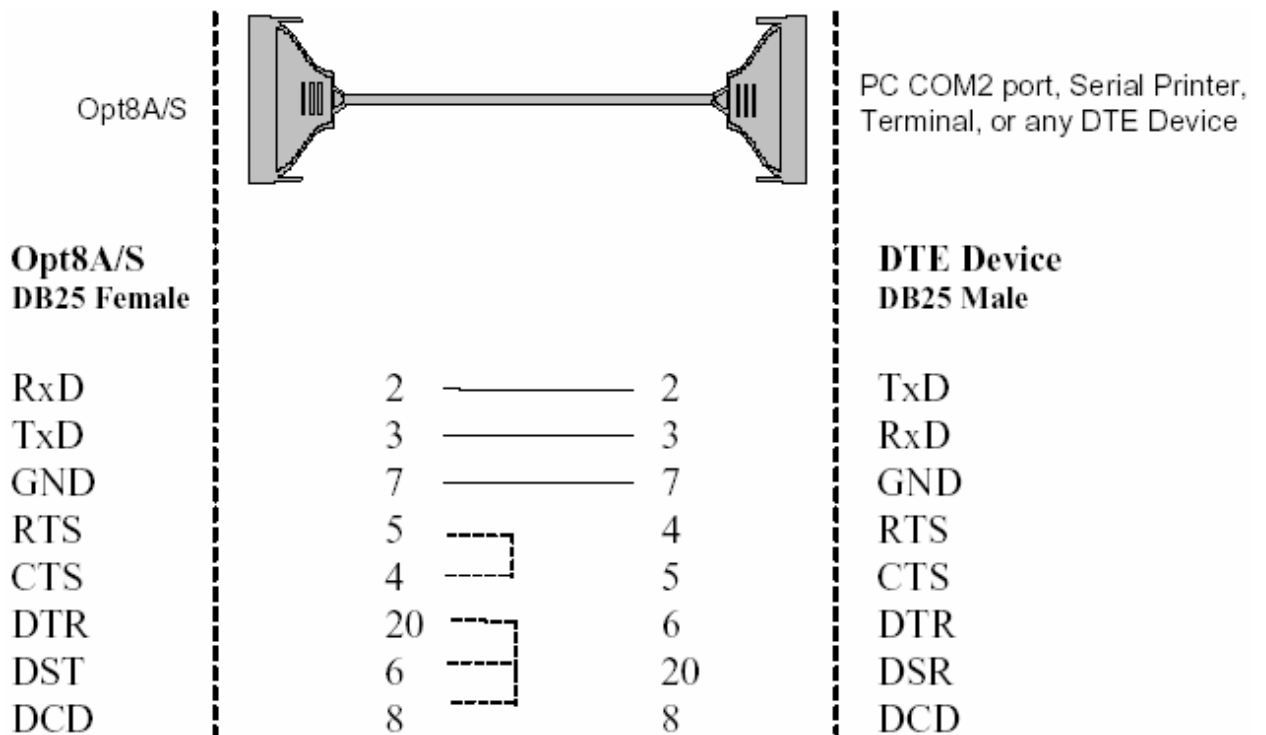


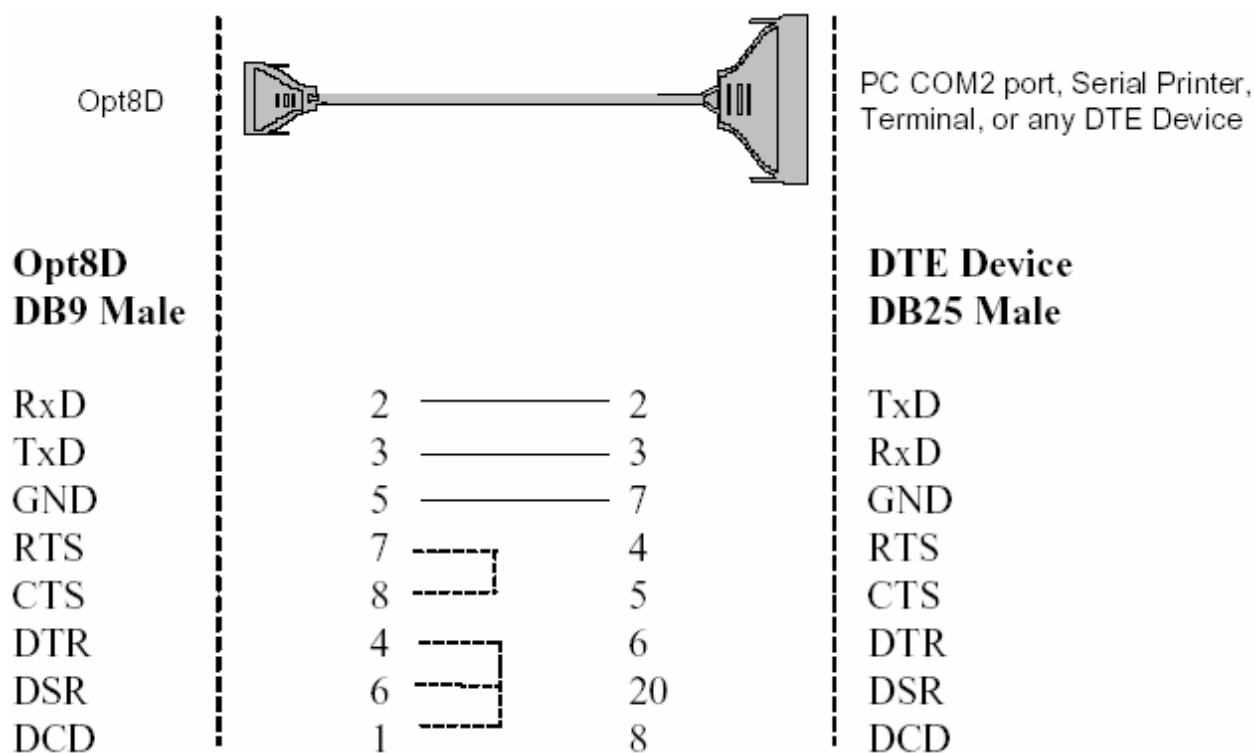


**Тип 3: Соединение Smartio C168 с устройством DTE по 3-х проводному интерфейсу.**

Если параметр [Hardware flow control] установлен в положение "ON", вы должны соединить (или закоротить) RTS с CTS и DSR с DTR, DCD со стороны MOXA, как показано на следующих рисунках. Если

параметр [**Hardware flow control**] установлен в положение “**OFF**”, вы можете оставить контакты RTS, CTS, DSR, DTR, DCD открытыми.





## Распайка разъемов RS-422 для Opt8J/F/Z

Для Smartio C168 разработаны следующие соединительные модули RS-422:

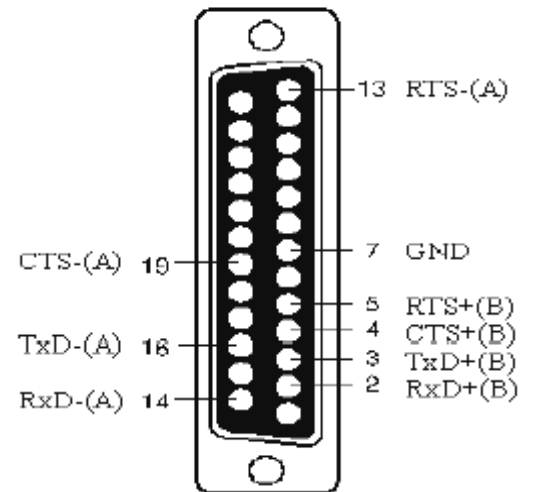
- Opt8J:** Соединительный модуль с 8 гнездами DB25. Установите переключатель в позицию **OFF** (RS-422) для нужного порта(ов).
- Opt8F:** Соединительный модуль с 8 гнездами DB25 и с оптической изоляцией (500V).
- Opt8Z:** Соединительный модуль с 8 гнездами RS-422 DB25, но без оптической изоляции.



## RS-422 Pinouts for Opt8J/F/Z:

Opt8J/F/Z	
2	RxD+(B)
3	TxD+(B)
14	RxD-(A)
16	TxD-(A)
7	GND
4	CTS+(B)
5	RTS+(B)
13	RTS-(A)
19	CTS-(A)

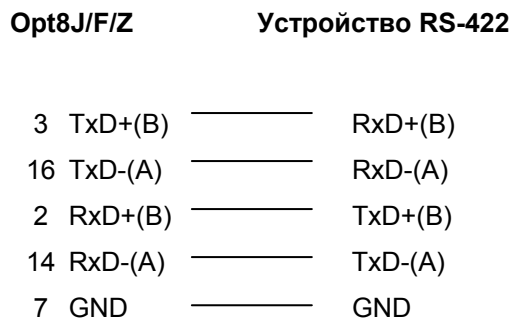
} Opt8J only



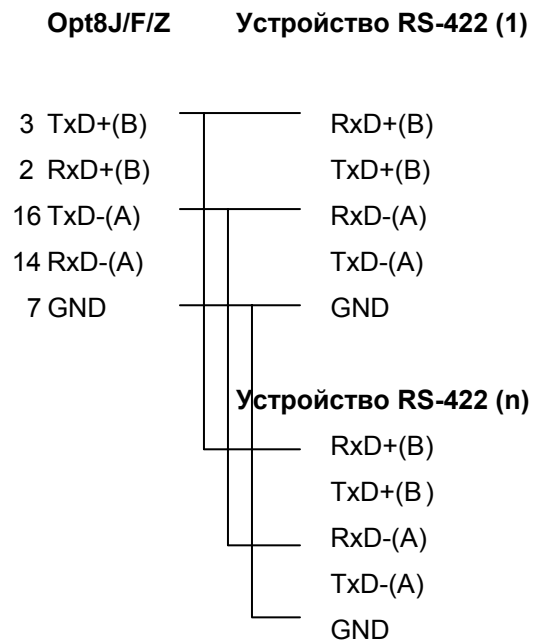
Длина линии RS-422 может достигать 4000 футов (1200 метров). Эти модули нуждаются во внешнем источнике питания (5V DC). В комплекте поставляется источник питания на напряжение 110V или 220V AC.

Далее приводятся основные схемы включения для RS-422:

### Rs-422 Point-to-Point



### RS-422 Boardcasting



## Opt8J with Handshaking

---

	OptJ		Устройство RS-422
3	TxD+(B)	—————	RxD+(B)
16	TxD-(A)	—————	RxD-(A)
2	RxD+(B)	—————	TxD+(B)
14	RxD-(A)	—————	TxD-(A)
7	GND	—————	GND
5	RTS+(B)	—————	CTS+(B)
13	RTS-(A)	—————	CTS-(A)
4	CTS+(B)	—————	RTS+(B)
19	CTS-(A)	—————	RTS-(A)

## Распайка разъёма RS-485 для Opt8J

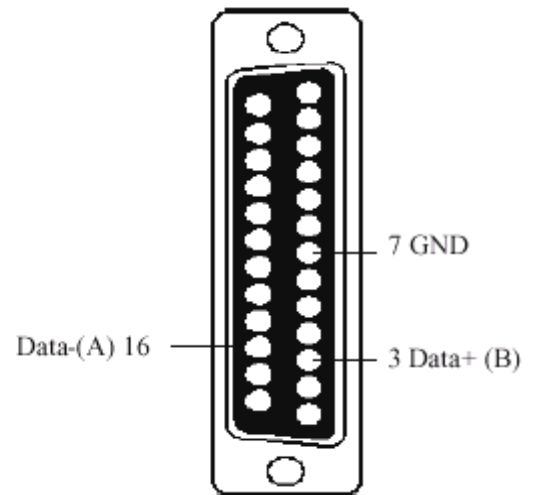
Для плат серии Smartio C168 разработан следующий соединительный модуль RS-485:

**Opt8J:** Соединительный модуль RS-422/485 с 8 гнездами DB25.  
Установите переключатель в позицию **ON** (RS-485) для  
нужного порта(ов).

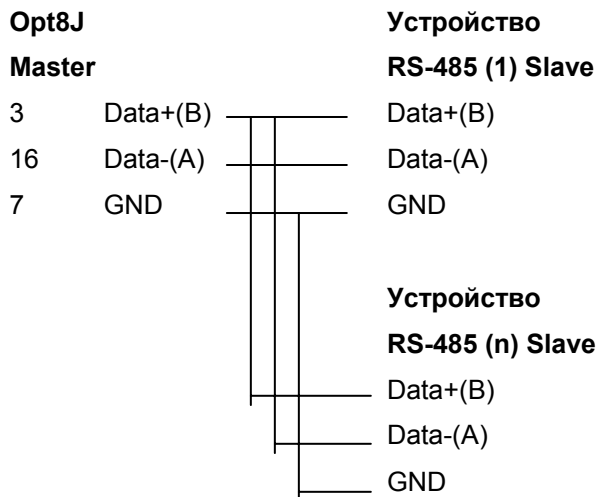
Opt8J поддерживает **только 2-х проводную полудуплексую RS-485 связь**. **Ввод/вывод** данных осуществляется с помощью одних и тех же контактов в зависимости от сигнала RTS.

## RS-485 Pinouts for Opt8J:

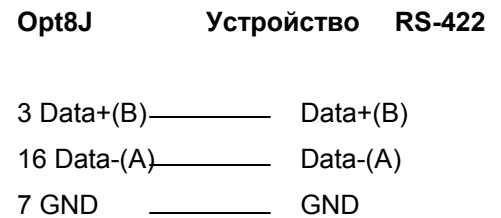
Opt8J	
3	Data+(B)
7	GND
16	Data-(A)



### Multidrop RS-485 Half-duplex



### Point-to-point RS-485 Half-duplex



Для ознакомления с подробностями программирования Opt8J RS-485 см. раздел **“Программирование RS-485”** главы **“Программное обеспечение”**.

## RS-422/485 Согласование импедансов

Для последовательной связи RS-422/485, когда электрический сигнал перемещается в линии передачи через соединения с разными сопротивлениями, несоответствие импедансов иногда становится причиной возникновения отраженного сигнала. Отраженный сигнал может вызвать искажение, которое приведет к возникновению ошибок

связи. Решение заключается в том, чтобы согласовать импедансы на разных концах линии с помощью оконечных резисторов.

Значение оконечных резисторов должно равняться волновому сопротивлению линии передачи. Резисторы должны быть добавлены **с принимающей стороны.**

	Opt8J/F/Z		Устройство RS-422/485
	3	TxD+(B)	↕ RxD+(B)
	16	TxD-(A)	↕ RxD-(A)
	2	RxD+(B)	↕ TxD+(B)
	14	RxD-(A)	↕ TxD-(A)
Только для Opt8J	5	RTS+(B)	↕ CTS+(B)
	13	RTS-(A)	↕ CTS-(A)
	4	CTS+(B)	↕ RTS+(B)
	19	CTS-(A)	↕ RTS-(A)

**Обратите внимание:**

1. ↕ - оконечный резистор, устанавливаемый с принимающей стороны.
2. Рекомендуемый оконечный резистор для кабеля AWG#26 - 100 ом.
3. Рекомендуемый оконечный резистор для телефонного кабеля - 600 ом.

## 6. Решение проблем

Далее рассматриваются наиболее распространенные проблемы, возникающие при работе с платой серии Smartio C168H и возможные пути их решения. Если проблема не решится одним из описанных ниже способов, то обратитесь за помощью к вашему дилеру или МОХА. Для осуществления максимально быстрой технической поддержки, перешлите вашему дилеру "**Отчет о проблеме**", содержащийся в конце руководства.

### Решение общих проблем

1. **В процессе инсталляции драйвера плата МОХА не обнаруживается драйвером МОХА.**

#### **Аппаратные причины и решения**

- a. Плата не установлена или плохо закреплена. Пожалуйста, установите ее.
- b. Плата неправильно подключена к системе. Если это так, то переставьте плату в 16-битный слот ISA. Иногда слот, в который вставляется плата, оказывается неисправным. В этом случае, пожалуйста, попробуйте использовать другой слот.

2. **Плата и драйвер МОХА активизированы, но передача / получение данных не осуществляется.**

#### **Аппаратные причины и решения**

- a. Проверьте правильность подключения кабеля; при необходимости обратитесь к главе "**Соединительные модули (Opt8x) и распайка разъемов**".
- b. Кабель или плата неисправны. Для проверки Вы можете использовать другой порт, соединительный модуль или плату. Утилита PCComm "Diagnostic" под Windows NT и Windows 95/98

также позволяет проверить работоспособность плат MOXA и состояние портов. Если сообщение диагностики показывает ошибку, то замените соответствующий компонент.

### **Программные причины и решения**

- a. Smartio C168H проверяет состояние линии (CTS) прежде, чем поступают данные, если в настройках или используемой программе управление RTS/CTS потоком данных установлено в позицию "Enable". Для правильного монтажа кабеля см. главу **"Соединительные модули (Opt8x) и распайка разъемов"**; также проверьте состояние линии подозрительного порта, используя для диагностики светодиодные индикаторы рядом с разъёмом.
- b. Возможно, приложение, управляющее платой, написано неправильно согласно соответствующему API операционной системы. Для проверки установите заведомо работоспособное приложение или утилиту, поставляемую MOXA, например, PComm "Terminal Emulation" или "HyperTerminal" под Windows NT и Windows 95/98..

### **3. Почему утилита IO-IRQ под DOS не может добавить плату в конфигурацию?**

В основе этой проблемы может лежать несколько причин:

- a. Пользователь не знает или забыл CAP адрес. Смотрите проблему **4**, где будет указано ее решение.
- b. CAP адрес платы конфликтует с адресами ввода-вывода других плат. Пожалуйста, измените адреса ввода-вывода других плат, чтобы устранить конфликт.
- c. Плата Smartio C168 установлена в неправильный или плохой слот. Переставьте ее в исправный слот ISA.
- d. Плата Smartio C168 неисправна. Необходим ремонт.

Если установленные платы имеют по умолчанию САР адрес 0x180, адрес ввода-вывода первого порта, то для устранения конфликта можно сделать следующее:


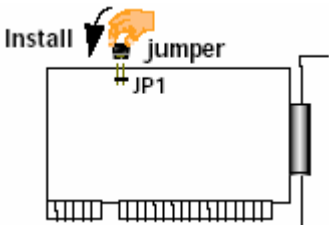
- a. Установите перемычку JP1 в левом верхнем углу платы. При этом САР адрес примет значение 0x700.
- b. Измените адреса ввода-вывода существующих плат.



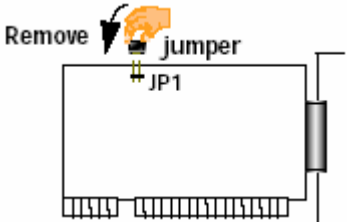

#### 4. Что делать, если пользователь не знает или забыл САР адрес платы Smartio C168?

Так как плата Smartio C168 не имеет переключателей и перемычек, то конфигурирование параметров осуществляется только в утилите lo-irq.exe под DOS.

Чтобы настроить плату, вам необходимо знать САР адрес платы, т. к. через этот канал утилита lo-irq.exe осуществляет доступ к аппаратным установкам платы.

Далее следуют инструкции, как поступить, если пользователю не известен САР адрес.

Шаг 1	Выключите компьютер.	
Шаг 2	Установите на плату перемычку.	

Шаг 3	Включите компьютер. Теперь САР адрес имеет значение 0xA700.	
Шаг 4	Запустите утилиту io-irq под DOS.	
Шаг 5	Введите САР адрес, чтобы сделать плату доступной. <i>Введите САР адрес в шестнадцатиричной форме:: A700.</i>	
Шаг 6	Теперь вы увидите предыдущие аппаратные настройки платы. <i>Теперь вы можете их изменить. Запомните САР адрес.</i>	
Шаг 7	Выйдите из утилиты IO-IRQ.	
Шаг 8	Выключите компьютер.	
Шаг 9	Снимите с платы перемычку.	
Шаг 10	Включите компьютер.	



# Windows NT

Этот раздел предлагает варианты решения проблем, возникающих под Windows NT. Для решения общих проблем, см. предыдущий раздел **"Решение общих проблем"**.

- 1. После перезагрузки системы в файле регистрации событий появляется сообщение "Another driver in the system which did not report its resources, has already claimed the interrupt used by xxx.)."**

Это свидетельствует о том, что плата MOXA обнаружена, но присвоенное ей IRQ конфликтует с другим устройством. В этом случае сначала проверьте установки BIOS, а затем задайте значение доступного IRQ.

- 2. После системной перезагрузки в файле регистрации появляется сообщение об ошибке "Cannot find any configured MOXA Smartio/Industio series board!"**

- Некоторые сетевые платы конфликтуют с нашей платой.  
Пожалуйста, избегайте использования адреса ввода-вывода 0x300 для других плат.
- Проверьте аппаратные настройки платы Smartio C168 в утилите Io-irq.exe. Затем необходимо проверить соответствие аппаратных настроек, включая адреса ввода-вывода для каждого порта, вектор прерывания, IRQ, с программными настройками драйвера.
- Адрес ввода-вывода может конфликтовать с настройками других устройств. Задайте другое значение, к примеру, адрес ввода-вывода 0x280, вектор прерывания: 0x2C0.
- Плата неправильно установлена на плате. Убедитесь, что плата стоит в соответствующем слоте.
- Слот, в котором установлена плата, неисправен. В этом случае следует переставить плату в другой исправный слот.

f. Плата неисправна.

**3. Номера COM платы Smartio C168H и другого устройства конфликтуют между собой.**

Номера COM разных плат конфликтуют. Измените номер COM в настройках платы MOXA.

**4. Нестабильная работа системы Windows NT (синий экран).**

Возможная причина - конфликт IRQ или памяти с другими ISA адаптерами типа плат LAN и SCSI или системой BIOS. Для разрешения ситуации, пожалуйста, обратитесь к соответствующей проблеме в предыдущем разделе "**Решение общих проблем**".

## **Windows 95/98**

Этот раздел помогает решить проблемы, возникающие в системе Windows 95/98. Для решения общих проблем, см. предыдущий раздел "Решение общих проблем".

**1. Система не может обнаружить плату Smartio C168H board!**

**После перезапуска система появляется сообщение об ошибке "Smartio C168 Series (CAP=0x0180, port 1=COM3): Board is not found".**

a. Некоторые сетевые платы конфликтуют с нашей платой.

Пожалуйста, избегайте использования адреса ввода-вывода 0x300 для других плат.

b. Проверьте аппаратные настройки платы Smartio C168 в утилите io-irq.exe. Затем необходимо проверить соответствие аппаратных настроек, включая адреса ввода-вывода для каждого порта, вектор прерывания, IRQ, с программными настройками драйвера.

- c. Адрес ввода-вывода может конфликтовать с настройками других устройств. Задайте другое значение, к примеру, адрес ввода-вывода 0x280, вектор прерывания: 0x2C0.
- d. Плата неправильно установлена на плате. Убедитесь, что плата стоит в соответствующем слоте.
- e. Слот, в котором установлена плата, неисправен. В этом случае следует переставить плату в другой исправный слот.
- f. Плата неисправна.

## DOS

Этот раздел помогает решить проблемы, возникающие в системе DOS. Для решения общих проблем, см. предыдущий раздел "Решение общих проблем".

1. **После загрузки драйвера SER-DRV.EXE появляется ошибка "None serial port found!".**
  - a. Убедитесь, что используете правильный драйвер.
  - b. Проверьте, установлена ли плата в соответствующий ISA/EISA слот.
  - c. Проверьте соответствие аппаратных и программных настроек платы.

## UNIX

Этот раздел помогает решить проблемы, возникающие в системе UNIX. Для решения общих проблем, см. предыдущий раздел "Решение общих проблем".

1. **Когда в системе используется несколько портов TTY, могут появляться сообщения о следующих ошибках: "Time out table overflow", "File table overflow", "Region table overflow".**

Вышеупомянутые ошибки подразумевают, что системных ресурсов не достаточно. Вы должны настроить параметры ядра на большие значения и перекомпилировать ядро. Чтобы узнать подробности настройки параметров и компилирования ядра, обратитесь к справочнику по системе UNIX.

"Time out table overflow": слишком маленький параметр NCALL.  
"File table overflow": слишком маленький параметр NFILE или NINODE.  
"Region table overflow": слишком маленький параметр NREGION или NPROC.

## **2. Перекомпилированное ядро не загружается.**

Драйвер C168 неправильно встроен в новое ядро.

- a. Пожалуйста, используйте последнюю рабочую копию ядра, чтобы загрузиться снова. Резервная копия ядра: **/unix.moxa** для SCO UNIX и **/stand/unix.moxa** для UNIX SVR4.2.
- b. Затем удалите драйвер C168. Для удаления драйвера обратитесь к разделу "UNIX" главы "Инсталляция драйвера".
- c. Установите драйвер C168 еще раз.

# Приложение. Техническая информация

## Спецификация

- ❖ Интерфейс с компьютером: 16-разрядный ISA (EISA совместимый)
- ❖ Число портов: 8
- ❖ Адреса ввода-вывода: 0x0000~0xFFFF
- ❖ IRQ: 2, 3, 4, 5, 7, 10, 11, 12, 15
- ❖ Количество бит данных: 5, 6, 7, 8
- ❖ Количество стоп-бит: 1, 1.5, 2
- ❖ Четность: none, even, odd, space, mark
- ❖ Контроллер UART: 8
- ❖ Скорость (bps.): 50 ~ 921.6K
- ❖ Разъёмы: 8 X DB25 (штекер/гнездо)
- ❖ Используемые сигналы: RS-232: TxD, RxD, RTS, CTS, DTR, DSR, DCD, GND  
RS-422: TxD+(B), TxD-(A), RxD+(B), RxD-(A), GND; (Opt8J: RTS+(B), RTS-(A), CTS+(B), CTS-(A))  
RS-485: Data+(B), Data-(A), GND
- ❖ Защита от выбросов: max 2000V (C168HS)
- ❖ Рабочая температура: 0 ~ 55° C
- ❖ Потребляемые токи: 180mA max. (+5V), 110mA max. (+12V), 160mA max. (-12V)
- ❖ Габариты: 157mm X 93mm
- ❖ Операционные системы: см. ниже список драйверов поддержки.

### Серия Smartio C168

Windows NT	+
Windows 95/98	+
DOS	+
SCO UNIX/OpenServer	+
UNIX SVR4.2	+
LINUX	R
SCO XENIX	C
QNX	C
FreeBSD	C

- + : драйвер MOXA поставляется с изделием
- R : драйвер MOXA поставляется после запроса
- C: драйвер поддерживается операционной системой

Обратите внимание: вы всегда можете скачать последнюю версию драйвера с FTP сервера MOXA

## UART 16C550C

Микросхема UART **16C550C** является интеллектуальным асинхронным контроллером, способным к поддержке одного полнодуплексного канала, который может передавать и принимать данные со скоростью до **921.6 Kbps** одновременно. Чтобы повысить максимальную производительность, в нем реализованы специальные интегрированные возможности FIFO и аппаратный контроль потока, уменьшающие число прерываний на центральном процессоре платы и предотвращающие любую потенциальную потерю данных.

## Карта адресов ввода-вывода

Ниже приводится таблица адресов ввода-вывода, которые необходимо учитывать при настройке портов платы Smartio C168 во избежание возникновения конфликтов.

I/O Address	Device
000-01F	DMA controller 1
020-03F	Interrupt controller
040-05F	Timer
060-06F	Keyboard
070-07F	Real-time clock DMA controller 1
080-09F	DMA page register
0A0-0BF	Interrupt controller 2
0C0-0DF	DMA controller
0F0-0FF	Math coprocessor
100-1EF	Not usable
1F0-1F8	Fixed disk
200-207	Game I/O
278-27F	Parallel printer port 2 ( LP2: )
2F8-2FF	Serial Port 2 ( COM2: )
300-31F	Prototype card
360-36F	Reserved
378-37F	Parallel printer port 1 ( LP1: )
3B0-3BF	Monochrome display
3C0-3CF	Reserved
3D0-3DF	Color graphics display
3F0-3F7	Diskette controller
3F8-3FF	Serial port 1 ( COM 1: )

## Контакты разъема DB62

Далее следует распайка контактов разъема DB62 на задней планке.

№ контакта	Сигнал	№ контакта	Сигнал	№ контакта	Сигнал
1	TxD1	22	RxD1	43	CTS1
2	DTR1	23	DSR1	44	RTS1
		24	DCD1	45	GND
3	RxD2	25	TxD2	46	CTS2
4	DSR2	26	DTR2	47	RTS2
5	DCD2				
6	TxD3	27	RxD3	48	CTS3
7	DTR3	28	DSR3	49	RTS3
		29	DCD3	50	GND
8	RxD4	30	TxD4	51	CTS4
9	DSR4	31	DTR4	52	RTS4
10	DCD4	32	GND		
11	RxD5	33	TxD5	53	CTS5
12	DSR5	34	DTR5	54	RTS5
13	DCD5			55	GND
14	TxD6	35	RxD6	56	CTS6
15	DTR6	36	DSR6	57	RTS6
		37	DCD6	58	GND
16	RxD7	38	TxD7	59	CTS7
17	DSR7	39	DTR7	60	RTS7
18	DCD7	40	GND		
19	RxD8	41	TxD8	61	CTS8
20	DSR8	42	DTR8	62	RTS8
21	DCD8				



# Отчет о неисправностях

## Серия Smartio C168

Имя Заказчика	
Компания:	
Телефон:	Факс:
Электронная почта:	Дата:

1. Изделие МОХА:  Серия Smartio C168      Модель:  C168H     C168HS  
Серийный номер \_\_\_\_\_

2. Версия драйвера МОХА: \_\_\_\_\_

3. Аппаратные параметры настройки МОХА:

3.1. Пожалуйста, укажите аппаратную конфигурацию IO-IRQ.EXE под DOS или Windows 95/98  
DOS Promt.

Порт	1	2	3	4	5	6	7	8
Адрес ввода-вывода								
IRQ								

Вектор прерывания: \_\_\_\_\_

Скорость: \_\_\_\_\_ (High/Normal)

3.2. Переключатель JP1:  открыта       закрыта

4. Операционная система:  Windows 95       Windows 98  
 Windows NT 3.51     Windows NT 4.0  
 DOS       UNIX       Другие

5. Персональный компьютер: Изготовитель \_\_\_\_\_ Модель \_\_\_\_\_

6. Процессор: Скорость \_\_\_\_\_ МГц, Изготовитель \_\_\_\_\_, Модель \_\_\_\_\_

7. BIOS: Изготовитель \_\_\_\_\_, Версия \_\_\_\_\_

8. Описание проблемы: пожалуйста, максимально подробно опишите имеющиеся симптомы, включая сообщения об ошибках. Мы будем следовать вашему описанию, чтобы воспроизвести проблему.

Плата не найдена.       Плата найдена, но не может передавать данные.  
 Данные передаются, но теряются.     Данные передаются, но с искажениями.  
 Другие варианты. Рекомендуется сделать детальное описание сообщений об ошибках:

# Процедура возвращения

Для ремонта, обмена или возврата изделия, Вы должны:

- ❖ Предъявить гарантийный талон.
- ❖ Максимально подробно заполнить прилагаемую анкету.
- ❖ Получить расписку о получении товара от коммерческого представителя или дилера.
- ❖ Тщательно упаковать изделие в неэлектризующийся пакет и с предоплатой переслать его дилеру.

## По вопросам продаж и поддержки обращайтесь:

Архангельск +7 (8182) 45-71-35	Кемерово +7 (3842) 21-56-70	Новосибирск +7 (383) 235-95-48	Сочи +7 (862) 279-22-65
Астрахань +7 (8512) 99-46-80	Киров +7 (8332) 20-58-70	Омск +7 (381) 299-16-70	Ставрополь +7 (8652) 57-76-63
Барнаул +7 (3852) 37-96-76	Краснодар +7 (861) 238-86-59	Орел +7 (4862) 22-23-86	Сургут +7 (3462) 77-96-35
Белгород +7 (4722) 20-58-80	Красноярск +7 (391) 989-82-67	Оренбург +7 (3532) 48-64-35	Тверь +7 (4822) 39-50-56
Брянск +7 (4832) 32-17-25	Курск +7 (4712) 23-80-45	Пенза +7 (8412) 23-52-98	Томск +7 (3822) 48-95-05
Владивосток +7 (4232) 49-26-85	Липецк +7 (4742) 20-01-75	Пермь +7 (342) 233-81-65	Тула +7 (4872) 44-05-30
Волгоград +7 (8442) 45-94-42	Магнитогорск +7 (3519) 51-02-81	Ростов-на-Дону +7 (863) 309-14-65	Тюмень +7 (3452) 56-94-75
Екатеринбург +7 (343) 302-14-75	Москва +7 (499) 404-24-72	Рязань +7 (4912) 77-61-95	Ульяновск +7 (8422) 42-51-95
Ижевск +7 (3412) 20-90-75	Мурманск +7 (8152) 65-52-70	Самара +7 (846) 219-28-25	Уфа +7 (347) 258-82-65
Казань +7 (843) 207-19-05	Наб.Челны +7 (8552) 91-01-32	Санкт-Петербург +7 (812) 660-57-09	Хабаровск +7 (421) 292-95-69
Калуга +7 (4842) 33-35-03	Ниж.Новгород +7 (831) 200-34-65	Саратов +7 (845) 239-86-35	Челябинск +7 (351) 277-89-65
			Ярославль +7 (4852) 67-02-35

сайт: [moxa.pro-solution.ru](http://moxa.pro-solution.ru) | эл. почта: [mxk@pro-solution.ru](mailto:mxk@pro-solution.ru)  
телефон: 8 800 511 88 70